Multilevel Monte Carlo Methods in Atmospheric Dispersion Modelling

Katsiolides, Grigoris

Award date:
2018

Awarding institution:
University of Bath

Multilevel Monte Carlo Methods in Atmospheric Dispersion Modelling

Grigoris Katsiolides

A thesis submitted for the degree of Doctor of Philosophy

University of Bath

Department of Mathematical Sciences

October 2018

Declaration of any previous submission of the work

The material presented here for examination for the award of a higher degree by research has not been incorporated into a submission for another degree.

. . . . . .

Grigoris Katsiolides

Declaration of authorship

I am the author of this thesis, and the work described therein was carried out by myself personally.

. . . . . .

Grigoris Katsiolides

Summary

This thesis considers the application of Monte Carlo and improved timestepping methods to stochastic Lagrangian models currently used by the Met Office in Atmospheric Dispersion modelling. These models are given in the form of stochastic differential equations (SDEs) and are used to predict the spread and transport of atmospheric pollutants, such as volcanic ash, in operational forecasting, emergency response situations and scientific research. The Met Office currently uses the Standard Monte Carlo (StMC) algorithm with the Symplectic Euler timestepping method. Although the StMC method can be fast enough when a low accuracy is required, it quickly becomes expensive for higher accuracies. Another difficulty is that the Symplectic Euler method requires small timesteps to be stable.

To improve the methods currently used by the Met Office we consider the Multilevel Monte Carlo (MLMC) algorithm (Giles, 2008) which has shown great potential for reducing the asymptotic computational complexity when compared to StMC in many applications. In particular, we study both numerically and theoretically, by using modified equations analysis (Shardlow, 2006), (Zygalakis, 2011), (Müller et al., 2015), the application of the complexity theorem (Giles, 2008) which describes in which cases MLMC can reduce the asymptotic cost rate. Together with this application, we develop and implement a new algorithm for the treatment of reflective boundary conditions while simulating dispersion in the atmospheric boundary layer which gives the correct variance decay rate required by the complexity theorem. In order to reduce the overall cost we consider improved timestepping methods based on a splitting approach. With the use of a Lyapunov function and the theory from Khasminskii (2011) and Milstein and Tretyakov (2005) we also show existence and uniqueness of solutions for our models and convergence of timestepping methods.

Our theoretical results and numerical methods initially apply on one-dimensional models. Later, we extend our numerical methods to higher-dimensional models where we also study the effect of a background velocity field.

Acknowledgements

I would like to thank my supervisors Eike Mueller, Robert Scheichl and Tony Shardlow who guided and advised me during my PhD. Their expertise, encouragement and support made this thesis possible and equipped me with great knowledge and skills.

I would also like to thank my industrial supervisor at the Met Office, David Thomson, for all the useful discussions and comments over the last few years.

Contents

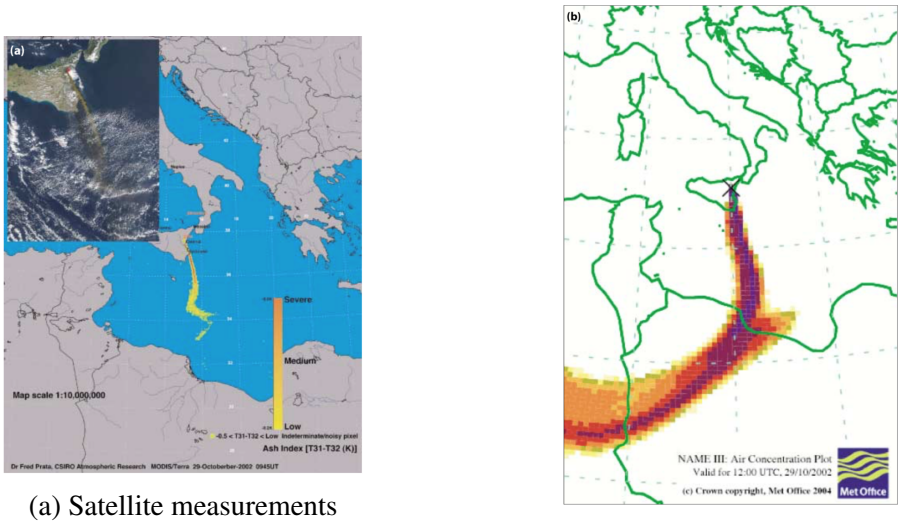
Chapter 1

Introduction

1.1 The subject of the thesis

Atmospheric dispersion modelling is the area of science that studies the spread and transport of atmospheric pollutants with the help of suitable mathematical models and algorithms. Currently the Met Office uses the atmospheric dispersion model NAME (Numerical Atmospheric - dispersion Modelling Environment) (Jones, 2004), (Jones et al., 2007) for modelling the transport and dispersion of atmospheric pollutants such as volcanic ash or radioactive particles and also for routine air quality forecasts and scientific research. It was originally designed to study the spread of radioactive particles after the Chernobyl disaster in 1986 (Smith and Clark, 1989). Some more recent applications of NAME include the study of the impact from the Fukushima nuclear accident in 2011 (Leadbetter et al., 2015) and the study of the spread of volcanic ash after the volcanic eruptions in Iceland in 2010 (Dacre et al., 2011), (Webster et al., 2012). This list of practically relevant examples, clearly shows the importance of NAME and in general why atmospheric dispersion modelling is an interesting and very important problem. All emergency response applications require fast NAME model predictions.

NAME is a Lagrangian Dispersion model, i.e. it tracks a number of model particles through a turbulent atmosphere flow field by solving a Stochastic Differential Equation (SDE). The distribution of those model particles provides a Monte Carlo estimate for the solution of the underlying SDE and the Standard Monte Carlo (StMC) method is used to approximate the value of some quantity of interest such as the mean particle position or the particle's concentration. An example of an output field generated by



(a) Satellite measurements

(b) NAME simulation

Figure 1-1: The eruption of Etna, Sicily in 2002 (Jones, 2004)

NAME can be seen in Figure 1-1 which presents the modelling of the eruption of Etna in Sicily in 2002 and compares it with satellite measurements. The images are from Jones (2004).

Monte Carlo methods (Glasserman, 2003), (Lemieux, 2009) have been widely used in various applications in areas such as mathematics, physics, biology and finance. Although the method had already appeared in some form centuries ago (eg Buffon's Needle, 18th century), its systematic development was made in the mid-1940's by John von Neumann and Stanislaw Ulam and its name first appeared in Metropolis and Ulam (1949) (Lemieux, 2009). The method's main advantages rely on the simplicity of its use and the fact that the cost does not grow exponentially with dimension as with the cost from a Partial Differential Equation (PDE) approach. For example we could use Finite Elements or Finite Differences to solve the Fokker-Planck PDE which corresponds to the SDE models but this approach is affected by the curse of dimensionality and therefore becomes expensive very quickly when higher-dimensional models are studied. For a three-dimensional spatial model, the PDE contains six variables which depend on time and this causes a large increase in the cost.

In its simplest form, the StMC provides approximations of solutions by simply taking the average over a large number of simulations. As explained later in the context of Stochastic Differential Equations (SDEs), the root mean square error ϵ of an approximation is split into the discretisation and sampling errors which contribute a

factor of ϵ^{-1} and ϵ^{-2} to the cost respectively. The total cost then has an asymptotic order of $\epsilon^{-1} \cdot \epsilon^{-2} = \epsilon^{-3}$ which becomes very large as ϵ decreases. In the literature, there are plenty of variance reduction techniques for example antithetic variables, control variates and importance sampling (Glasserman, 2003), (Lemieux, 2009) which can reduce the cost of the method but without improving the asymptotic cost rate. In order to reduce the cost rate, we consider the Multilevel Monte Carlo (MLMC) method (Heinrich, 2001), (Giles, 2008) and its application to models used by the Met Office in atmospheric dispersion modelling.

Although for a low accuracy the StMC method can be fast enough, by increasing the accuracy it becomes very expensive and slow to run since the cost rate has an asymptotic order of ϵ^{-3}. Therefore, a more efficient method is required. The MLMC method has been successfully applied in various areas such as mathematical finance (Giles, 2008), (Giles and Szpruch, 2012), partial differential equations with random coefficients (Cliffe et al., 2011) and uncertainty quantification in subsurface flows (Graham et al., 2015). The method showed its great potential of reducing the asymptotic order of the cost to ϵ^{-2} which makes it extremely important in many applications. In particular, it can be very important to the Met Office since in an emergency responce situation, like for example a volcano eruption, it is necessary to produce fast and accurate predictions.

One of the key properties that lead to the cost rate reduction is the fact that MLMC constructs a hierarchy of levels with larger timestep sizes and also computes estimates of the differences between two estimators which have a smaller variance. Then, under certain conditions, it can reduce the number of samples required to achieve a certain tolerance ϵ on the root mean square error and also shift the cost on the coarsest levels which are cheaper to compute since they use a larger timestep. In all methods that we studied, the bias of the SDE timestepping methods depends linearly on the timestep size h, i.e. bias $= C \cdot h$ and the total cost is affected by the constant of proportionality C. At the Met Office they currently use the Symplectic Euler timestepping method so it becomes important to study more advanced methods which have a smaller bias constant and therefore reduce the total cost.

In this thesis we study atmospheric dispersion models used by the Met Office which describe the evolution of the particle's position $\mathbf{X}(t)$ and particle's turbulent velocity component $\mathbf{U}(t)$ in the atmosphere. These models are given in d spatial dimensions

and have the general SDE form

$$d\mathbf{U}(t) = -\Lambda\left(\mathbf{X}(t)\right)\mathbf{U}(t)dt - \nabla_{\mathbf{X}}V\left(\mathbf{X}(t),\mathbf{U}(t)\right)dt + \Sigma\left(\mathbf{X}(t)\right)d\mathbf{W}(t), \tag{1.1}$$

$$d\mathbf{X}(t) = \left(\mathbf{U}(t) + \boldsymbol{v}\left(\mathbf{X}(t),t\right)\right)dt, \tag{1.2}$$

where $\mathbf{X}(t), \mathbf{U}(t), \boldsymbol{v}\left(\mathbf{X}(t),t\right) \in \mathbb{R}^d$, $V\left(\mathbf{X}(t),\mathbf{U}(t)\right) \in \mathbb{R}$, $\Lambda\left(\mathbf{X}(t)\right)$ and $\Sigma\left(\mathbf{X}(t)\right)$ are diagonal $d \times d$ matrices and $\mathbf{W}(t)$ is a d-dimensional Brownian motion (see Definition 2.1). The notation $\nabla_{\mathbf{X}}$ means that we take the gradient vector only with respect to the variable $\mathbf{X}$. $\boldsymbol{v}\left(\mathbf{X}(t),t\right)$ is the background velocity field, $\Lambda\left(\mathbf{X}(t)\right)$ contains velocity memory terms, $\nabla_{\mathbf{X}}V\left(\mathbf{X}(t),\mathbf{U}(t)\right)$ guarantees the well-mixed condition (Thomson, 1987) and $\Sigma\left(\mathbf{X}(t)\right)$ is the stochastic component.

The functions $\Lambda\left(\mathbf{X}(t)\right)$, $V\left(\mathbf{X}(t),\mathbf{U}(t)\right)$, $\Sigma\left(\mathbf{X}(t)\right)$ and $\boldsymbol{v}\left(\mathbf{X}(t),t\right)$ depend on the value of d and we will consider separately the three cases when $d = 1, 2, 3$. When $d = 1$ the model will describe one-dimensional vertical dispersion, when $d = 2$ it will describe a two-dimensional horizontal dispersion and when $d = 3$ it will describe a three-dimensional dispersion that combines the vertical and horizontal structures of the first two cases. When $d = 1$ and $d = 3$ we assume that the motion of the particles is constrained to a boundary layer of fixed height and as we describe later, some of the SDE coefficients in (1.1) and (1.2) will have singularities at the boundary points. Note that the boundary layer is the part of the atmosphere where turbulence plays a significant role and hence the SDE which models this by a random term is applicable.

The main subject of the thesis is the implementation and theoretical analysis of the MLMC method to produce numerical approximations of the above model with an improved asymptotic rate of the order ϵ^{-2} as discussed above. For this, various timestepping methods must be explored that will enable us to discretise our models, reduce the total cost (by reducing the constant in the cost rates) and deal with any singularities present in the SDE coefficients in order to produce particle trajectories. In addition to specifying the particles' initial position and velocity it is necessary to add suitable boundary conditions at the bottom and top of the boundary layer in the one- and three-dimensional setups. Since the particle trajectories themselves are not directly measurable, we are interested in approximating the value of $\mathbb{E}\left[\phi(\mathbf{X}(t))\right]$ for some quantity of interest $\phi : \mathbb{R}^d \to \mathbb{R}$ that describes a physical property. The quantity of interest ϕ can be simple and continuous, like for example the identity function which gives the mean particle position but it can also be discontinous, like the indica-

tor function which gives the particle concentration. The latter can be used to measure the volcanic ash present in certain regions of the atmosphere after a volcanic eruption which is very important for air travelling.

Monte Carlo and timestepping methods are analysed both numerically and theoretically. With the design of more efficient algorithms we compare the performance of Standard and Multilevel Monte Carlo and apply alternative timestepping methods with higher stability and study possible improvements. With suitable assumptions we study the problem theoretically which will help in the improvement of the current algorithms used by the Met Office by identifying the conditions under which our methods perform better. In Section 1.2 we give a more detailed description of the aims and challenges of this thesis.

1.2 The aims of the thesis

The central aim of this thesis is the design of a Monte Carlo algorithm and more advanced timestepping methods for SDEs in atmospheric dispersion modelling that will improve the StMC-Symplectic Euler combination currently used by the Met Office. In particular, we are interested in implementing the MLMC method that will help us to improve the asymptotic cost rate from ϵ^{-3} to ϵ^{-2}. The overall cost, i.e. the constant in the above cost rates, depends on the timestepping method and reducing this constant is also very important especially in physical applications which usually do not require a small ϵ. To achieve this goal several difficulties need to be overcome.

Firstly, some of the coefficients in the SDE equations (1.1) and (1.2) have singularities at the end points of our boundary layer which can lead to severe limitations on the timestep size and limit the use of MLMC. To make sure that our timestepping methods remain stable we must deal with these singularities by either removing them with the help of some regularisation or by implementing more advanced methods which are less affected by them. In addition, our choice of timestepping methods must be justified by theoretical results which show that the SDE approximations converge to the exact solution.

Next, we study under which conditions the MLMC method gives the improved cost rate of ϵ^{-2} since this is not always the case. Our approach is the theoretical application of the complexity theorem (Giles, 2008) which requires the correct coupling of random variables on the coarse levels. Each timestepping method requires a different

coupling treatment and this application can be supported by both numerical and theoretical results where this is possible. For the theoretical application we use modified equations analysis (Shardlow, 2006), (Zygalakis, 2011), (Müller et al., 2015) which is an alternative approach to traditional strong approximation results.

Another challenge is the correct treatment of the boundary conditions at the top and bottom of the atmospheric boundary layer. In this thesis we consider simple elastic reflection. As we will see later if the random variables on subsequent levels are not coupled correctly to account for reflection, then the MLMC variance increases and violates one of the conditions in the complexity theorem (Giles, 2008) which is required to achieve the optimal ϵ^{-2} cost rate. Therefore, we need to develop an algorithm that contains the proper coupling of random variables when the particles are reflected.

Also, we need to be careful that our quantity of interest does not increase the variance of the methods. One of the most important quantities considered in this thesis is the indicator function which gives an approximation of the particle concentration. The indicator function however is not continuous which causes an increase in the variance and slows down the MLMC method. Therefore, it is important to find a suitable smooth approximation that will reduce the variance and at the same time allow us to minimise any extra approximation errors.

All the above topics are first studied for a one-dimensional spatial model for vertical dispersion, i.e. equations (1.1) and (1.2) with $d = 1$ and also $\boldsymbol{v}(\mathbf{X}(t), t) = 0$. Later, higher dimensional models are explored where we also study the effect of a non-zero background velocity field $\boldsymbol{v}(\mathbf{X}(t), t)$. In particular, we study a two-dimensional model for horizontal dispersion and a three-dimensional model that combines the first two.

Finally, based on the outcome of this thesis the methods that we study can then be implemented in the Met Office's NAME dispersion model. The ultimate goal is to use MLMC and the new timestepping methods to improve the performance of the NAME model. This is important for many reasons like for example providing faster and more accurate results in emergency responce situations (eg volcano eruptions).

1.3 The main achievements

The main achievements of this thesis are:

(i) Successfully implemented MLMC in one-dimensional atmospheric dispersion modelling and produced numerical results which clearly show the efficiency of

the MLMC over the StMC method for an inhomogeneous turbulence model, especially for small tolerances. The results were produced using the C++ code initially developed for the paper Müller et al. (2015) and which we later enriched for the purposes of this thesis.

(ii) Studied possible improvements on the performance of Monte Carlo from more stable and accurate timestepping methods. In particular, we compare the Symplectic Euler method currently used by the Met Office with two improved methods, the Geometric Langevin (Bou-Rabee and Owhadi, 2010) and BAOAB (Leimkuhler and Matthews, 2015) which are based on an SDE splitting approach. We observed that timestepping methods based on a splitting approach can reduce the StMC cost when applied to one-dimensional models with varying turbulence profiles. As a result, their combination with StMC can be a better choice for large tolerance errors.

(iii) Developed and implemented a new algorithm for the treatment of reflective boundary conditions which preserves the quadratic variance decay of the MLMC method (Katsiolides et al., 2018). The new method is based on extending the initial model with reflection to a model without reflection which results in the correct coupling of random variables in the MLMC coarse steps.

(iv) Applied the smoothing polynomial technique as in Giles et al. (2015) to deal with the discontinuity of the indicator function in the concentration problem. With this approximation we improve the MLMC variance rates and therefore the performance.

(v) Proved existence and uniqueness of a regular solution of the inhomogeneous one-dimensional model using a Lyapunov function and the theory in Khasminskii (2011). Using the same Lyapunov function and the theory from Milstein and Tretyakov (2005) we also show that under some assumptions our timestepping methods converge weakly.

(vi) Proved theoretically that the complexity theorem (Giles, 2008) holds for the one-dimensional models with suitable regularisation. For a simplified model representing homogeneous turbulence we apply a direct approach based on strong approximation results. For our inhomogeneous model we apply a method based

on the theory of modified equations (Shardlow, 2006), (Zygalakis, 2011), (Müller et al., 2015) which provides an alternative approach to the strong approximation results.

(vii) Implemented an adaptive MLMC (Giles et al., 2016) algorithm for one-dimensional models and observed some small improvements for the Symplectic Euler method.

(viii) Implemented StMC and MLMC for two- and three-dimensional models. We observed that for higher dimensional models the MLMC method improves the asymptotic cost rate when compared to StMC. Also, we demonstrated that in some cases the BAOAB method can reduce the overall cost of StMC.

1.4 The structure of the thesis

In Chapter 2 we introduce our one-dimensional models and describe the physical interpretation of the basic variables. We describe the model scales by using dimensional analysis, define the boundary conditions and discuss how we deal with the singularities in the turbulence profiles present in the models. Then, we talk about the quantities of interest that we are going to approximate, compute the systems invariant measure and discuss some relations with molecular dynamics.

In Chapter 3 we present the timestepping methods that we use to discretise the one-dimensional models and define the Standard and Multilevel Monte Carlo methods in the context of SDEs. We review the Symplectic Euler method currently used by the Met Office and also define two more advanced methods, the Geometric Langevin and BAOAB which are based on SDE splitting methods. We then describe how we measure the Monte Carlo approximation errors and the strategy we follow in order to achieve a certain error tolerance. Also, we derive a theoretical bound for the StMC cost rate as a function of the error tolerance and review the complexity theorem (Giles, 2008) which explains when the MLMC method has a better cost rate when compared to StMC.

In Chapter 4 we present some theoretical results about the one-dimensional models. We begin by proving that our basic inhomogeneous model has a unique solution by using a Lyapunov function and the theory in Khasminskii (2011). We then talk about the convergence of our timestepping methods and using the same Lyapunov function with

the theory from Milstein and Tretyakov (2005) we show that under certain conditions they are all weakly convergent to the exact solution.

In Chapter 5 we study the theoretical application of the complexity theorem (Giles, 2008) to the particular one-dimensional models in atmospheric dispersion modelling. In Section 5.1 we consider a simplified model and present a direct approach based on strong approximation results. In Section 5.2 we present a new algorithm for the correct treatment of reflective boundary conditions (Katsiolides et al., 2018). The basic idea is to extend the initial model to a problem without boundary conditions and count the number of reflections which then gives the correct coupling of random variables in the MLMC coarse levels. In Section 5.3 we study the conditions under which the complexity theorem holds for our basic inhomogeneous model using an alternative approach based on modified equations analysis (Shardlow, 2006), (Zygalakis, 2011), (Müller et al., 2015). Lastly, in Section 5.4 we describe how we deal with the discontinuity of the indicator function in the concentration problem by using the smoothing poynomial technique from Giles et al. (2015).

In Chapter 6 we present several numerical experiments for the one-dimensional models to compare the efficiency of Standard and Multilevel Monte Carlo with our timestepping methods. We study the mean particle position, concentration and density function; all results and conclusions in this section have been published in Katsiolides et al. (2018). In all the experiments we use a uniform timestep size except from the last part of the chapter where we consider adaptive timestepping.

In Chapter 7 we study higher-dimensional models. We construct a realistic background velocity field (i.e. the function $\boldsymbol{v}(\mathbf{X}_t, t)$) in equation (1.2)) with the correct energy spectrum (Nastrom and Gage, 1985) and a realistic wind-shear due to the Ekman spiral (Holton, 2004). We then discuss the corresponding model scales, the particle spread plots and finally present numerical results for two- and three-dimensional models. For the numerical results we consider only one quantity of interest, the particle concentration.

In the Conclusion, we summarise all the achievements, suggest some ideas for further research and finally in the Appendix we present the technical details of some of the results of this thesis.

Chapter 2

Atmospheric Dispersion Modelling

Atmospheric dispersion modelling studies the spread and transport of atmospheric pollutants and is important at the Met Office in many areas such as emergency response situations (volcano eruptions, nuclear accidents, smoke from large fires etc) and routine forecasting applications. To model the dispersion, the three dimensional total velocity field $\mathbf{u}(\mathbf{X}, t)$, which depends on both the spatial coordinate $\mathbf{X}$ and time t, is first split into two components. The first component corresponds to the large-scale flow $\boldsymbol{v}(\mathbf{X}, t)$ which is a deterministic velocity field and is given from the Met Office's forecast model. The second component corresponds to the unresolved turbulence $\mathbf{U}(\mathbf{X}, t)$ which can be modelled by a stochastic velocity field. Combining these two quantities we can then write

$$\mathbf{u}(\mathbf{X}, t) = \boldsymbol{v}(\mathbf{X}, t) + \mathbf{U}(\mathbf{X}, t). \tag{2.1}$$

The velocity field $\boldsymbol{v}(\mathbf{X}, t)$ has the property that

$$\boldsymbol{v}(\mathbf{X}, t) = \langle \mathbf{u}(\mathbf{X}, t) \rangle, \tag{2.2}$$

where $\langle \cdot \rangle$ denotes the ensemble average. The ensemble average of $\mathbf{U}(\mathbf{X}, t)$ is zero since as we see later, the turbulence is assumed to be Gaussian with mean zero. From George (2009) the ensemble average of an observable quantity A which depends on the particle position and velocity, is defined by

$$\langle A \rangle = \lim_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} a^{(n)}, \tag{2.3}$$

where $a^{(n)}$ is the n^{th} realisation of A, i.e. the value of A obtained at the n^{th} independent and identical repetition of the experiment.

At the Met Office, particle dispersion is studied by using the NAME Lagrangian Dispersion model. Lagrangian dispersion models describe the transport of a passive tracer particle in a turbulent velocity field in the form of an SDE. For a given particle, the Lagrangian velocity is the velocity at a given time t and each particle follows the mean flow $\boldsymbol{v}(\mathbf{X}, t)$ to which we then add a random velocity component $\mathbf{U}(t)$. More specifically, $\mathbf{U}(t)$ denotes the turbulent component of the Lagrangian velocity and at equilibrium, i.e. after a large travel time, it agrees in distribution with the turbulent field $\mathbf{U}(\mathbf{X}, t)$ for particles which pass through point $\mathbf{X}$ at time t (Katsiolides et al., 2018). A general SDE in d spatial dimensions for a Lagrangian dispersion model (later simply called d-dimensional) can be written as

$$d\mathbf{U}(t) = -\Lambda(\mathbf{X}(t))\mathbf{U}(t)dt - \nabla_{\mathbf{X}}V(\mathbf{X}(t), \mathbf{U}(t))dt + \Sigma(\mathbf{X}(t))d\mathbf{W}(t), \tag{2.4}$$

$$d\mathbf{X}(t) = (\mathbf{U}(t) + \boldsymbol{v}(\mathbf{X}(t), t))dt, \tag{2.5}$$

where $\mathbf{X}(t), \mathbf{U}(t), \boldsymbol{v}(\mathbf{X}(t), t) \in \mathbb{R}^d$, $V(\mathbf{X}(t), \mathbf{U}(t)) \in \mathbb{R}$, $\Lambda(\mathbf{X}(t))$ and $\Sigma(\mathbf{X}(t))$ are diagonal $d \times d$ matrices and $\mathbf{W}(t)$ is a d-dimensional Brownian motion. According to this model, the particles follow the mean velocity $\boldsymbol{v}(\mathbf{X}(t), t)$, called background velocity, to which we add the random variable $\mathbf{U}(t)$ that represents turbulence. We also assume that the particles are independent, non-interacting and of unit mass. The following definition of Brownian motion can be found in Lemieux (2009).

Definition 2.1. *A standard one-dimensional Brownian motion is a continuous-time stochastic process* $\{W(t), t \geq 0\}$ *with the following properties:*

1. *W(0) = 0.*
2. *The increments over disjoint intervals are independent, i.e. for* $r < s < t < u$ *we have that* $W(u) - W(t)$ *and* $W(s) - W(r)$ *are independent.*
3. *The increments are stationary, i.e. for any* $r, s, t > 0$ *we have that* $W(r + t) - W(r)$ *and* $W(s + t) - W(s)$ *have the same probability density function, which is normal with mean* 0 *and variance* t*.*

A d-dimensional Brownian motion $\mathbf{W}(t)$ *is a vector of d independent Brownian motions* $W(t)$*.*

Note that we could also consider a deterministic model by solving the corresponding Fokker-Planck Partial Differential Equation (also called Kolmogorov's Forward Equation) which is equivalent to solving the SDE (Øksendal, 2003). For the SDEs (2.4) and (2.5) it is given by

$$\frac{\partial \rho}{\partial t} = -\sum_{i=1}^{d} \left[\frac{\partial}{\partial X_i} \left[(U_i + v_i(\mathbf{X}, t))\, \rho \right] + \frac{\partial}{\partial U_i} \left[\left(-\Lambda_{ii}(\mathbf{X}) U_i - \frac{\partial V}{\partial X_i}(\mathbf{X}, \mathbf{U}) \right) \rho \right] \right] + \frac{1}{2} \sum_{i=1}^{d} \frac{\partial^2}{\partial U_i^2} \left[\left(\Sigma_{ii}(\mathbf{X}) \right)^2 \rho \right], \tag{2.6}$$

where $\rho(\mathbf{X}, \mathbf{U}, t)$ is the probability density of the particles in the phase space defined by their position and velocity. The index i denotes the i^{th} component of the corresponding variable. We could then use Finite Elements or Finite Differences to approximate the solution. This approach however is affected by the curse of dimensionality since we have $2d$ number of variables which depend on time and this gives a further advantage to the Monte Carlo method whose cost does not grow exponentially with dimension. Here we consider only Monte Carlo methods applied to SDEs and the comparison with the Fokker-Planck approach can be a topic for future research.

For simplicity, and since in the atmosphere the strongest variations usually occur in the vertical direction, we firstly consider one-dimensional models which means that we study equations (2.4) and (2.5) with $d = 1$. These models describe vertical dispersion and we also set $\boldsymbol{v}(X, t) = 0$ (where X is the vertical coordinate) since the vertical background velocities are much smaller than the horizontal velocities and can be neglected. Later, in Chapter 7 we present the generalisation to higher dimensions.

This chapter is structured as follows. Firstly, we introduce our basic one-dimensional model for inhomogeneous turbulence and describe the equations' coefficients and variables. Then, we introduce a simplified version of the first model which describes dispersion in a homogeneous turbulence field. Next, we describe the model scales by using dimensional analysis, discuss how we deal with the singularities present in the model and present our choice of boundary conditions. Finally, we briefly discuss the quantities of interest that we are going to approximate and also describe some connections to molecular dynamics. Here we also compute the system's invariant measure.

2.1 The one-dimensional Inhomogeneous turbulence model

Our basic one-dimensional model for vertical dispersion assumes stationary (independent of time) and inhomogeneous (dependent of position) turbulence and is given by the following SDE

$$dU(t) = -\lambda(X(t))U(t)dt - \frac{\partial V}{\partial X}(X(t), U(t))dt + \sigma(X(t))dW(t), \tag{2.7}$$

$$dX(t) = U(t)dt. \tag{2.8}$$

In this equation, t is the time, $U(t)$ is the Lagrangian vertical velocity fluctuation due to turbulence, $X(t)$ is the vertical position and $W(t)$ is a Brownian motion. The coefficients $\lambda(X)$, $\frac{\partial V}{\partial X}(X, U)$ and $\sigma(X)$ are given in terms of the two functions $\sigma_U(X)$, $\tau(X)$ and are related by

$$\lambda(X) = \frac{1}{\tau(X)}, \quad \sigma(X) = \sqrt{\frac{2\sigma_U^2(X)}{\tau(X)}}, \tag{2.9}$$

$$\frac{\partial V}{\partial X}(X, U) = -\frac{1}{2}\left[1 + \left(\frac{U}{\sigma_U(X)}\right)^2\right]\frac{\partial \sigma_U^2}{\partial X}(X). \tag{2.10}$$

With this notation, the function $V(X, U)$ can also be written as

$$V(X, U) \equiv -\frac{1}{2}\left[\sigma_U^2(X) + U^2 \log\left(\sigma_U^2(X)/\sigma_U^2(X_{\text{ref}})\right)\right], \tag{2.11}$$

where X_{ref} is a fixed reference height. $\sigma_U^2(X)$ is the variance of the turbulent component of the background velocity field and $\tau(X)$ is the Lagrangian local velocity decorrelation timescale. The form of these functions will be discussed in Section 2.2. Although the velocity $U(t)$ depends on time we call the model stationary because $\tau(X)$ and $\sigma_U(X)$ are independent of t. Similarly, we call the model inhomogeneous because $\tau(X)$ and $\sigma_U(X)$ depend on X. We assume that the motion of the particles is restricted to a boundary layer of height H, i.e. their vertical position can only vary between 0 and H (see also Figure 2-2).

Next, we need to specify suitable initial conditions $(X(0), U(0))$ which can either be deterministic or stochastic depending on the problem. For the one-dimensional

model we assume that both quantities are deterministic. For higher-dimensional models however, and since the background velocity is non-zero we assume that the initial velocity adapts to the ambient velocity of the flow when the particles are released. Practically this means that the initial position is still deterministic but the initial velocity is random with each component independent and normally distributed with mean zero and variance $\sigma_U^2(\mathbf{X}(0))$ (for higher-dimensional models this function can be different in each direction).

Equation (2.7) was derived in Thomson (1987) where the author presents several criteria for the selection of these stochastic Lagrangian models. These criteria are also reviewed in Rodean (1996). One of the most important, is the well-mixed condition (Thomson, 1987) that gives rise to the coefficient $\frac{\partial V}{\partial X}(X(t), U(t))$ and requires that particles which are initially uniformly mixed to stay uniformly mixed. In mathematical terms this means that the stationary distribution of the particle density $\rho(X, U)$ of $(X(t), U(t))$ has two basic properties. The first property is that the marginal distribution of X (i.e. $\rho(x) = \int_{-\infty}^{\infty} \rho(x, u)du$) is uniform in space and the second property is that conditioned on X we have that $\rho(u|x = X)$ has the same distribution as the unresolved turbulence $U(X, t)$ which is assumed to be Gaussian (Katsiolides et al., 2018). In Lemma 2.3 we verify that this is indeed the case by showing that the stationary distribution is given by

$$\rho(X, U) = \frac{1}{\sqrt{2\pi}\sigma_U(X)} \exp\left(-\frac{U^2}{2\sigma_U^2(X)}\right), \tag{2.12}$$

which also requires that the function $V(X, U)$ has the form given by equation (2.11). Other criteria include the correct behaviour of the small-time velocity distribution of particles from a point source and the consistency of the forward and reverse dispersion formulations. However, as it is proved in Thomson (1987) these criteria follow from the well-mixed condition.

2.1.1 Simplification to homogeneous turbulence

The main numerical results that we present for the one-dimensional case are for the inhomogeneous model that we have just described. However, for global dispersion applications and for long time scales, a simplified form of equation (2.7) can be used in the approximations. This simplified model is obtained by assuming that $\tau(X)$ and

$\sigma_U(X)$ in equation (2.7) are constants which gives

$$dU(t) = -\lambda U(t)dt + \sigma dW(t), \tag{2.13}$$

$$dX(t) = U(t)dt, \tag{2.14}$$

where $\lambda = \frac{1}{\tau}$ and $\sigma = \sqrt{\frac{2\sigma_U^2}{\tau}}$. Since τ and σ_U are now independent of X the model describes homogeneous turbulence. A review of its derivation can be found in Rodean (1996).

All variables are as in the inhomogeneous model but now τ is equal to the timescale for the Lagrangian velocity autocorrelation of $U(t)$. In Taylor (1921) cited in Rodean (1996) the following equations were introduced for the timescale τ when the turbulence is homogeneous:

$$\tau = \int_{t_0}^{\infty} R(t - t_0)dt, \tag{2.15}$$

with

$$R(t - t_0) = \frac{\langle U(t)U(t_0)\rangle}{\langle U^2(t_0)\rangle}, \tag{2.16}$$

where t_0 is a reference time and $\langle\cdot\rangle$ denotes the ensemble average. In Taylor (1921) it is noted that the limiting form of a series expansion for the correlation coefficient R in equation (2.16) gives

$$R(t - t_0) = e^{-(t-t_0)/\tau}, \tag{2.17}$$

and therefore we can see that the autocorrelation decays exponentially with a characteristic timescale τ. Equation (2.17) can also be verified using the solution of equation (2.13) (See Appendix A.1).

The only results that we present for the homogeneous model are in Section 5.1 where we prove theoretically that the MLMC method performs with a better cost rate. Note that the simplified model has an analytic solution that we compute in Appendix A.2 where we also present some analysis of the spread of the particles.

2.1.2 Model scales and dimensional analysis

Before computing any numerical approximation involving our SDE models it is necessary to non-dimensionalise all the parameters since our code's input parameter file cannot contain any physical units. This dimensional analysis allows the physical interpretation of the numerical results by introducing suitable model scales.

For a physical quantity α we consider a reference scale α^{ref} and write

$$\alpha = \alpha^{\text{ref}}\hat{\alpha}, \tag{2.18}$$

where $\hat{\alpha}$ is the corresponding dimensionless parameter. In particular, for our model we consider a reference length X^{ref}, a reference speed U^{ref}, a reference time T^{ref} and write

$$X(t) = X^{\text{ref}}\hat{X}(t), \quad U(t) = U^{\text{ref}}\hat{U}(t), \quad t = T^{\text{ref}}\hat{t}, \tag{2.19}$$

where $\hat{X}(t)$, $\hat{U}(t)$ and $\hat{t}$ are the corresponding dimensionless quantities. Substituting in equation (2.8) gives

$$X^{\text{ref}}d\hat{X}(t) = U^{\text{ref}}T^{\text{ref}}\hat{U}(t)d\hat{t}, \tag{2.20}$$

so the reference variables are simplified and give the same equation as with the physical parameters provided

$$X^{\text{ref}} = U^{\text{ref}}T^{\text{ref}}. \tag{2.21}$$

In the following, we drop the hats from these variables and use dimensionless quantities in the equations, under the assumption that they measure the physical quantities in the units defined by X^{ref}, U^{ref} and T^{ref}. We choose $X^{\text{ref}} = 1000m$ (since our boundary layer's height is $1km$) and $U^{\text{ref}} = 1m/s$ which gives $T^{\text{ref}} = 1000s = 16\frac{2}{3}min$. As an example of how we use these scales in practise suppose that we are interested in releasing particles from an initial height of $50m$ with an initial velocity of $0.1m/s$. This simply means that we set $X(0) = 0.05$ and $U(0) = 0.1$.

2.2 The functions $\sigma_U(X)$ and $\tau(X)$

As discussed in Section 2.1 our inhomogeneous model contains the functions $\sigma_U(X)$ and $\tau(X)$ and their exact form depends on the atmospheric conditions of the boundary layer which is assumed to be of height H. In Webster et al. (2003) several profiles for $\sigma_U(X)$ and $\tau(X)$ are proposed which take into account stability conditions in the atmosphere. As a representative choice, we use the following form of $\sigma_U(X)$ as in Webster et al. (2003) for neutral and stable conditions

$$\sigma_U(X) = \kappa_\sigma u^* \left(1 - \frac{X}{H}\right)^{\frac{3}{4}}, \quad X \in (0, H), \tag{2.22}$$

where u^* is the friction velocity and κ_σ some dimensionless constant. For $\tau(X)$ we use

$$\tau(X) = \kappa_\tau \frac{X}{\sigma_U(X)}, \quad X \in (0, H), \tag{2.23}$$

for some dimensionless constant κ_τ which agrees with Wilson et al. (2009) where the expression $\tau(X) \approx 0.5X/\sigma_U$ is derived for a neutrally stable surface layer and horizontally homogeneous turbulence. Although there is some uncertainty in the exact form of these profiles, exploring more cases is beyond the scope of this thesis so we only focus on the above representative case. In our numerical experiments we use $\kappa_\sigma = 1.3$, $\kappa_\tau = 0.5$, $u^* = 0.2$ and $H = 1$ as in Cook (2013).

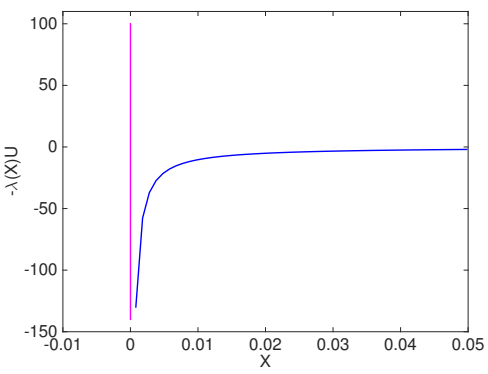
In equations (2.22) and (2.23) we can easily observe that $\sigma_U(X)$ and $\tau(X)$ have singularities or they are zero at the boundaries $X = 0$ and $X = H$ and as a result some of the SDE coefficients in (2.7) diverge near the boundaries. In Figure 2-1 we plot the SDE coefficients $-\lambda(X)U$, $-\frac{\partial V}{\partial X}(X, U)$ and $\sigma(X)$ and we can see that as X tends to zero or $H(= 1)$ some of the functions tend to infinity.

With these singularities it now becomes hard to simulate any trajectories and establish well-posedness since when the particles reach the boundaries we obtain infinite values which make some timestepping methods unstable. In order to deal with the singularities we regularise the functions $\tau(X)$ and $\sigma_U(X)$ by setting them to constant values below a height ϵ_{reg} and above a height $H - \epsilon_{\text{reg}}$. More specifically, we use the same regularisation as the Met Office by setting

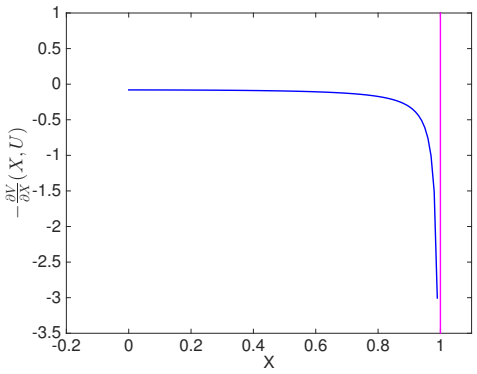
$$\tau(X) = \tau(\epsilon_{\text{reg}}), \quad \sigma_U(X) = \sigma_U(\epsilon_{\text{reg}}) \ \text{ if } \ X < \epsilon_{\text{reg}}, \tag{2.24}$$

$$\tau(X) = \tau(H - \epsilon_{\text{reg}}), \quad \sigma_U(X) = \sigma_U(H - \epsilon_{\text{reg}}) \ \text{ if } \ X > H - \epsilon_{\text{reg}}, \tag{2.25}$$

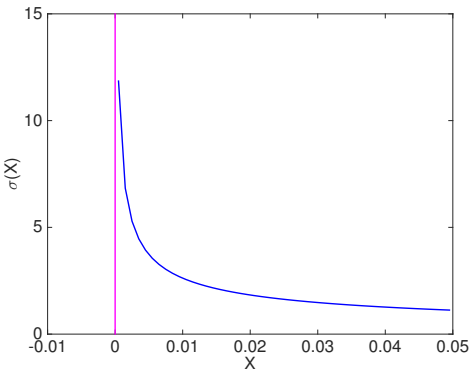
where $0 < \epsilon_{\text{reg}} \ll H$ is some small regularisation constant. Consequently, the functions $\tau(X)$ and $\sigma_U(X)$ can never be evaluated at the points $X = 0$ or $X = H$ which create the singularities and they are bounded with $\tau(\epsilon_{\text{reg}}) \leq \tau(X) \leq \tau(H - \epsilon_{\text{reg}})$ and $\sigma_U(H - \epsilon_{\text{reg}}) \leq \sigma_U(X) \leq \sigma_U(\epsilon_{\text{reg}})$. Currently at the Met Office $\tau(X)$ is kept constant below the height at which the physical value of $\tau(X)$ equals 20 sec. Using this property and the parameter values of $\tau(X)$ given above gives approximately $\epsilon_{\text{reg}} = 0.01$ which corresponds to a physical height of $10m$. The effect of ϵ_{reg} on the numerical results is studied in Section 6.4.1.



(a) $-\lambda(X)U$



(b) $-\frac{\partial V}{\partial X}(X,U)$



(c) $\sigma(X)$

Figure 2-1: SDE Coefficients for $U = 0.2$

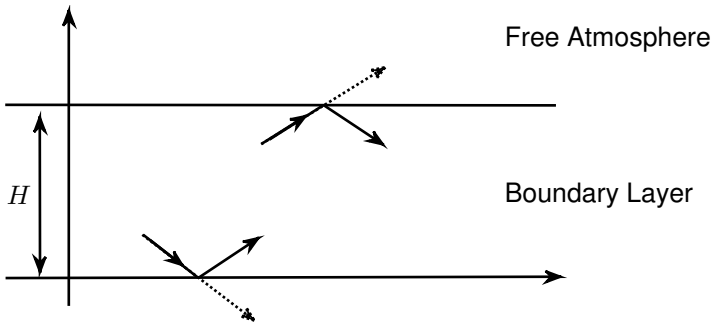


Figure 2-2: Reflective boundary conditions for a one-dimensional model.

2.3 Boundary conditions

In order to model the dispersion of particles we split the atmosphere in two parts: the boundary layer close to the ground and the free atmosphere. Our one-dimensional models apply on the boundary layer and this means that we restrict the motion of particles to the interval $[0, H]$. To keep the particles inside this layer and prevent them from going below ground or entering the free atmosphere above, it is necessary to add boundary conditions.

A realistic scenario for the boundary conditions at $X = 0$ would be to consider two possibilities. The particle is either absorbed at the boundary, so that it just stays at the ground, or it is reflected with a suitable probability for each case. Here for simplicity we consider only simple reflection so that when a particle hits the lower boundary it is reflected elastically as shown in Figure 2-2. In the real atmospheric conditions a particle can leave the boundary layer and enter the free atmosphere but here we also reflect at the upper boundary for simplicity.

The elastic reflective boundary conditions we use are mathematically defined in a similar way as in Bernardo et al. (2008). Suppose that at time t^{refl} a particle hits the boundary. If it hits the lower boundary ($X = 0$) then we use

$$\begin{aligned} \lim_{s\to 0, s>0} X(t^{\text{refl}} + s) &= - \lim_{s\to 0, s>0} X(t^{\text{refl}} - s), \\ \lim_{s\to 0, s>0} U(t^{\text{refl}} + s) &= - \lim_{s\to 0, s>0} U(t^{\text{refl}} - s), \end{aligned} \tag{2.26}$$

and if it hits the upper boundary ($X = H$) we use

$$\begin{aligned} \lim_{s\to 0, s>0} X(t^{\text{refl}} + s) &= 2H - \lim_{s\to 0, s>0} X(t^{\text{refl}} - s), \\ \lim_{s\to 0, s>0} U(t^{\text{refl}} + s) &= - \lim_{s\to 0, s>0} U(t^{\text{refl}} - s). \end{aligned} \tag{2.27}$$

Note that the condition on $X(t)$ is unnecessary as it follows from continuity, but we still include it because it will be helpful in the implementation of the boundary conditions in the discretisation methods discussed in section 3.1. More information about boundary conditions can be found in Rodean (1996) where the author presents a short review of various approaches found in the literature and how they interact with some of the model's quantities like for example the velocity decorrelation timescale $\tau(X)$.

Note that at any point in time the total mass is obtained by summing the masses of all model particles. Since in our model the mass of an individual particle does not change over time, mass conservation follows trivially if no trajectories are terminated and this is the case for the non-absorptive reflective boundary conditions considered here.

In Section 5.2 we discuss in more detail the effect of the boundary conditions in the performance of the MLMC method. As we will see, when reflection is not treated correctly it reduces the efficiency of the MLMC method and we will provide a new technique for the correct coupling of random variables in the presence of reflective boundary conditions which does not reduce the performance. This new technique will be based on extending the domain of X to the whole real line and make the problem equivalent to the one without boundary conditions.

Finally, we note that with the choice of reflective boundary conditions and under some assumptions on the functions $\sigma_U(X)$ and $\tau(X)$ it is possible to show existence and uniqueness of a solution to the inhomogeneous model. In Section 4.1 we include the details of the proof which is based on the existence of a Lyapunov function and the theory from Khasminskii (2011).

2.4 Quantities of interest

While the numerical solution of the SDE will give a set of paths, our main interest is to approximate the mean value of some functional ϕ of the path $(X(t), U(t))$ which we call the quantity of interest. In particular, we study the mean particle position $\mathbb{E}[X(T)]$

and the concentration $\mathbb{E}\left[\mathbb{1}_{[a,b]}(X(T))\right]$ for some $T > 0$ and $a, b \in \mathbb{R}$. The indicator function in the concentration problem is defined by

$$\mathbb{1}_{[a,b]}(x) = \begin{cases} 1 & \text{if} \quad x \in [a,b], \\ 0 & \text{otherwise.} \end{cases}$$

We first study the mean particle position because it is the simplest case to consider and then we study the concentration which is very important in atmospheric dispersion and of particular interest to the Met Office. As we will see later in Section 6.3, we can use a vector of indicator functions to construct a piecewise constant approximation to the concentration field at any given time T. For example, this would allow predicting the concentration of volcanic ash that a plane may encounter on different flight levels.

2.5 Relations with molecular dynamics

To put our work into a wider context, we now discuss some relations between our one-dimensional models and other areas of modelling and in particular molecular dynamics. The SDE equations (2.4) and (2.5) have the form of a well-known class of equations called the Langevin equations, which are also used for modelling in molecular dynamics. If we denote by $\mathbf{X}$ and $\mathbf{U}$ the particles' position and momentum vectors respectively then the simplified form of SDEs

$$d\mathbf{U}(t) = -\lambda \mathbf{U}(t)dt - \nabla G(\mathbf{X}(t))dt + \sigma d\mathbf{W}(t), \tag{2.28}$$

$$d\mathbf{X}(t) = \mathbf{U}(t)dt, \tag{2.29}$$

for some function G, constants λ and σ and Brownian motion $\mathbf{W}(t)$ (see Definition 2.1) are used to model a system of particles in a heat bath (Leimkuhler and Matthews, 2015). The dimension of $\mathbf{X}$ and $\mathbf{U}$ is now equal to the number of particles. The only interaction between the system of particles and the heat bath is the exchange of energy and when equilibrium is reached, the equilibrium distribution is given by

$$\tilde{\rho}(\mathbf{X}, \mathbf{U}) = Z^{-1} \exp\left(-\frac{2\lambda}{\sigma^2}\left(\frac{1}{2}\mathbf{U}^T\mathbf{U} + G(\mathbf{X})\right)\right), \tag{2.30}$$

where Z is a normalisation constant. Equation (2.30) is called Gibbs canonical distribution. The constants λ and σ are related with the system's temperature $\mathcal{T}$ with the equation

$$\kappa_B \mathcal{T} = \frac{\sigma^2}{2\lambda}, \tag{2.31}$$

where κ_B is Boltzmann's constant (Leimkuhler and Matthews, 2015).

In our inhomogeneous model equation (2.7), the derivative on the right hand side also depends on the velocity and this is due to the well-mixed condition (Thomson, 1987), (Rodean, 1996). The Gibbs canonical distribution equation (2.30) can motivate the choice of the invariant measure of our inhomogeneous model equations (2.7) and (2.8).

Definition 2.2. *The function $\rho(X, U, t)$ is called an invariant measure or equilibrium distribution for the SDEs* (2.7) *and* (2.8) *if it satisfies the Fokker-Planck equation* (2.6) *with $\frac{\partial \rho}{\partial t} = 0$ (Leimkuhler and Matthews, 2015).*

We give this measure in the following lemma.

Lemma 2.3. *An invariant measure of the system of equations* (2.7) *and* (2.8) *without boundary conditions is given by*

$$\rho(X, U) = \frac{1}{\sqrt{2\pi}\sigma_U(X)} \exp\left(-\frac{U^2}{2\sigma_U^2(X)}\right), \tag{2.32}$$

where $\sigma_U(X)$ is assumed to be sufficiently smooth.

$Proof$. From Definition 2.2 it is enough to show that equation (2.32) satisfies the corresponding Fokker-Planck equation with $\frac{\partial \rho}{dt} = 0$.

Using (2.6), the Fokker-Planck equation is given by

$$\frac{\partial \rho}{dt} = -\frac{\partial}{\partial u}\left[\left[-\frac{u}{\tau(x)} + \frac{1}{2}\left[1 + \frac{u^2}{\sigma_U^2(x)}\right]\frac{\partial \sigma_U^2(x)}{\partial x}\right]\rho\right] - \frac{\partial}{\partial x}(u\rho) + \frac{1}{2}\frac{\partial^2}{\partial u^2}\left[\frac{2\sigma_U^2(x)}{\tau(x)}\rho\right],$$

where $\rho(x, u, t)$ is the density of $(X(t), U(t))$ and simplifying gives

$$\begin{aligned}\frac{\partial \rho}{dt} = & \frac{\rho}{\tau(x)} + \frac{u}{\tau(x)}\frac{\partial \rho}{\partial u} - \sigma_U(x)\frac{\partial \sigma_U(x)}{dx}\frac{\partial \rho}{\partial u} - \frac{2u\rho}{\sigma_U(x)}\frac{\partial \sigma_U(x)}{\partial x} - \frac{u^2}{\sigma_U(x)}\frac{\partial \sigma_U(x)}{\partial x}\frac{\partial \rho}{\partial u} \\ & - u\frac{\partial \rho}{\partial x} + \frac{\sigma_U^2(x)}{\tau(x)}\frac{\partial^2 \rho}{\partial u^2}.\end{aligned} \tag{2.33}$$

Differentiating equation (2.32) gives

$$\frac{\partial \rho}{du} = -\frac{u\rho}{\sigma_U^2(x)}, \quad \frac{\partial^2 \rho}{du^2} = -\frac{\rho}{\sigma_U^2(x)} + \frac{u^2\rho}{\sigma_U^4(x)}, \quad \frac{\partial \rho}{dx} = -\frac{\rho}{\sigma_U(x)}\frac{\partial \sigma_U(x)}{\partial x} + \frac{u^2\rho}{\sigma_U^3(x)}\frac{\partial \sigma_U(x)}{\partial x},$$

and substituting in (2.33) we get

$$\frac{\partial \rho}{dt} = 0.$$

□

Lemma 2.3 completes our discussion about the well-mixed condition after we defined the inhomogeneous model in Section 2.1. As we can see from equation (2.32), the marginal distribution of X, $\rho(x) = \int_{-\infty}^{\infty} \rho(x,u)du = 1$, is uniform in space and the conditional distribution $\rho(u|x = X)$ is Gaussian with mean zero and variance $\sigma_U^2(X)$.

Lastly, from equation (2.31) we note that in the context of molecular dynamics $\sigma_U^2 = \frac{\sigma^2}{2\lambda} = \kappa_B \mathcal{T}$ and therefore, σ_U^2 can be seen as the system's "temperature".

However, the σ_U^2 that we are using in our one-dimensional models is not related to the air-temperature. It only describes measures of the fluctuation in the particle velocity due to turbulence and it is computed using data based on observations of velocity variances (Webster et al., 2003). Also, molecular diffusion is negligible compared to turbulent diffusion and the reason is that $D_{\text{molecular}} = \sigma^2_{U,\text{molecular}} \cdot \tau_{\text{molecular}}$ is much smaller than $D_{\text{turbulence}} = \sigma^2_{U,\text{turbulence}} \cdot \tau_{\text{turbulence}}$. The variable D_c where $c = \text{molecular, turbulence}$ is called the diffusion constant and after time $T \gg \tau_c$ the particles travel at a distance $\sqrt{D_c T}$ by diffusion. Since $D_{\text{molecular}}$ is much smaller we can then neglect molecular diffusion.

Chapter 3

Timestepping and Monte Carlo methods

Except for very special cases (such as the homogeneous turbulence model considered in Section 2.1.1) it is not possible to solve an SDE exactly. Instead, usually two approximations are made to obtain a numerical solution: (a) the total integration time is split into a finite number of small intervals and (b) expectation values (in the computation of quantities of interest) are replaced by Monte Carlo estimators. The stability and accuracy of both approximations need to be studied carefully.

In this chapter we define discrete timestepping methods and Standard and Multilevel Monte Carlo methods in the context of SDEs. Firstly, we present three timestepping methods called Symplectic Euler, Geometric Langevin and BAOAB which will enable us to produce particle trajectories of our SDE models. Then, we define Monte Carlo methods and describe how we measure the approximation errors. Also, we discuss how we choose the timestep size and the number of samples in order to achieve a particular tolerance on the total root mean square error. Finally, we compute the cost rate of StMC and we include the complexity theorem (Giles, 2008) that describes the conditions which guarantee a better asymptotic cost rate for the MLMC method.

3.1 Timestepping methods

To discretise our inhomogeneous model SDEs (2.7) and (2.8) we use three explicit timestepping methods. These methods are also extensions to SDEs of symplectic

methods for Hamiltonian systems. This means that when they are applied in a Hamiltonian system of Ordinary Differential Equations (ODEs) given by

$$\frac{dU}{dt} = -\frac{\partial H}{\partial X},$$
$$\frac{dX}{dt} = +\frac{\partial H}{\partial U},$$

for some function $H(X, U)$ (called the Hamiltonian) they preserve the system's Hamiltonian dynamics. In a Hamiltonian system the total area of the flow is preserved and if a timestepping method does not change this property is called symplectic (Hairer et al., 2006). The methods we use were chosen because the first one (Symplectic Euler) is very popular in operational models due to its simplicity and is currently used by the Met Office. The other two (Geometric Langevin, BAOAB) do not have any stability constraints on the timestep size as we discuss below (i.e. some restriction on the timestep size to make sure that the numerical approximation does not become large very quickly). More timestepping methods for SDEs, including the ones that we present here, can be found in Hairer et al. (2006), Kloeden and Platen (2011), Leimkuhler and Matthews (2015) and Müller et al. (2015). We define all timestepping methods in terms of the one-dimensional models and for higher dimensions we simply apply the same method on each direction.

In our numerical experiments we are interested in finding an approximation of the solution of the SDE models at time T so we integrate the SDE over the time interval $[0, T]$. Discretising in time, this interval will then be split into M subintervals of size $h = T/M$ and the exact solution $(X(t_n), U(t_n))$ is estimated at time $t_n = nh$ for each $n \in \{0, \ldots, M\}$. For each n the approximation is denoted by $(X_n, U_n) \approx (X(t_n), U(t_n))$.

The first timestepping method is the Symplectic Euler method (see for example Hairer et al. (2006)), currently used by the Met Office, and is given by

$$U_{n+1} = (1 - \lambda(X_n)h)\, U_n - \frac{\partial V}{\partial X}(X_n, U_n)h + \sigma(X_n)\sqrt{h}\xi_n, \tag{3.1}$$
$$X_{n+1} = X_n + U_{n+1}h, \tag{3.2}$$

where $\xi_n \sim Normal(0, 1)$ are independent and identically distributed (i.i.d.) Normal random variables. We observe that X_{n+1} is computed using U_{n+1} and this makes the

method symplectic for Hamiltonian systems (i.e. when $\lambda(x) = \sigma(x) = 0$ and V independent of U). Replacing U_{n+1} with U_n in equation (3.2) gives the Euler Maruyama method (see for example Hairer et al. (2006)) which is not symplectic. The Symplectic Euler method is stable if the timestep size h satisfies $|1 - \lambda(X_n)h| < 1$ which leads to the constraint $h < \frac{2}{\lambda(X_n)}$. As a result, for the solution to be stable, for large $\lambda(X)$ we have to choose smaller timesteps which increase the computational time and cost. In particular, for the inhomogeneous model, $\lambda(X)$ becomes large at the bottom of the boundary layer and this can lead to prohibitively tight bounds on the timestep size.

The other two timestepping methods, which do not have a stability issue are based on a splitting approach (Leimkuhler and Reich, 2004), (Leimkuhler and Matthews, 2015). For this, we write the SDE (2.7) in the form

$$dU(t) = dU^{(O)}(t) + dU^{(B)}(t), \tag{3.3}$$

$$\frac{dX(t)}{dt} = 0, \tag{3.4}$$

where

$$dU^{(O)}(t) = -\lambda(X(t))U^{(O)}(t)dt + \sigma(X(t))dW(t), \tag{3.5}$$

$$dU^{(B)}(t) = -\frac{\partial V}{\partial X}(X(t), U^{(B)}(t))dt. \tag{3.6}$$

with the accompanying velocity equation

$$dX(t) = U(t)dt. \tag{3.7}$$

Equations (3.5), (3.6) and (3.7) are discretised separately and one of the advantages of this method is that it may simplify a complicated problem by breaking it into a series of smaller and simpler problems. In addition, as we see below, the splitting methods that we consider have the very important property of not requiring any constraint on the timestep size in order to be stable.

In particular, when discretising equations (3.5) and (3.6) we assume that $X(t)$ is constant in the interval $[t_n, t_{n+1})$ (and equals X_n) and therefore we can solve equation (3.5) as an exact Ornstein-Uhlenbeck process over a single step. This removes the stability constraint and also improves the performance of the Monte Carlo method as demonstrated in Müller et al. (2015). As we will see, the term $\lambda(X_n)h$ will appear as

a negative exponent to the exponential function and when $\lambda(X_n)$ is large this expression will simply vanish without creating any stability problems as with the Symplectic Euler method. As we will see in Chapter 6 these methods reduce the bias error of the approximations. Following the notation from Leimkuhler and Matthews (2015) we refer to equation (3.5) as the O-update, to equation (3.6) as the B-update and to equation (3.7) as the A-update.

The first method based on this splitting is the Geometric Langevin method (Bou-Rabee and Owhadi, 2010) defined by

$$U^{\star}_{n+1} = e^{-\lambda(X_n)h}U_n + \sigma(X_n)\alpha_h\xi_n, \tag{3.8}$$

$$U_{n+1} = U^{\star}_{n+1} - \frac{\partial V}{\partial X}(X_n, U^{\star}_{n+1})h, \tag{3.9}$$

$$X_{n+1} = X_n + U_{n+1}h, \tag{3.10}$$

with $\alpha_h = \sqrt{\left(1 - e^{-2\lambda(X_n)h}\right)/2\lambda(X_n)}$ and $\xi_n \sim Normal(0,1)$ i.i.d. The method was introduced in the context of MLMC in Müller et al. (2015) and referred to as "Symplectic Euler/Ornstein - Uhlenbeck Splitting Method". The only difference in Müller et al. (2015) is that the potential function V depends just on X. With the notation from Leimkuhler and Matthews (2015) it can also be called OBA. This method was derived by first solving the Ornstein-Uhlenbeck process equation (3.5) over a single step and then by applying simple forward Euler steps on equations (3.6) and (3.7).

With this method there are no stability problems since when λ is large the exponential function simply disappears without any constraints on the choice of the timestep size. However, as we will see, when λ is large the multilevel variance increases and we discuss this for the simplified model in Section 5.1.

Note that for the simplified model where λ and σ_U are constants, letting $\lambda \to \infty$ in equations (3.8) and (3.9) gives

$$U_{n+1} = \sigma_U \xi_n. \tag{3.11}$$

This expression is also obtained by discretising the exact solution from Appendix A.2.1 and then letting $\lambda \to \infty$. This property is clearly not true for the Symplectic Euler method since by letting $\lambda \to \infty$ in equation 3.1 we obtain an infinite value.

The third and last timestepping method, which is also based on a splitting approach, is the Symmetric Langevin Velocity-Verlet method (Leimkuhler and Matthews, 2015)

defined by

$$U_{n+\frac{1}{2}} = U_n - \frac{\partial V}{\partial X}(X_n, U_n)\frac{h}{2}, \tag{3.12}$$

$$X_{n+\frac{1}{2}} = X_n + U_{n+\frac{1}{2}}\frac{h}{2}, \tag{3.13}$$

$$U^{\star}_{n+\frac{1}{2}} = e^{-\lambda(X_{n+\frac{1}{2}})h}U_{n+\frac{1}{2}} + \sigma(X_{n+\frac{1}{2}})\alpha_h(X_{n+\frac{1}{2}})\xi_n, \tag{3.14}$$

$$X_{n+1} = X_{n+\frac{1}{2}} + U^{\star}_{n+\frac{1}{2}}\frac{h}{2}, \tag{3.15}$$

$$U_{n+1} = U^{\star}_{n+\frac{1}{2}} - \frac{\partial V}{\partial X}(X_{n+1}, U^{\star}_{n+\frac{1}{2}})\frac{h}{2}, \tag{3.16}$$

where $\xi_n \sim Normal(0,1)$ i.i.d. and α_h is as above. Following the notation in Leimkuhler and Matthews (2015) we simply refer to this method as BAOAB. Notice that for both velocity and position the total update is of size h. Any other combination of these three updates with a total timestep of size h leads to different numerical schemes (Leimkuhler and Matthews, 2015).

All three timestepping methods converge with at least the same order of weak convergence as the Euler-Maruyama method does and we show this in Section 4.3.2 by using Taylor series approximations. Based on the numerical results from Chapter 6 we deduce that all methods are first order but when we analyse them and compute modified equations in Section 5.3.2 we see that BAOAB is very close to being a second order method and this has an effect on the performance of the StMC method. We briefly discuss strong error convergence at the beginning of Section 4.3.

3.2 Monte Carlo methods

We now define the Standard and Multilevel Monte Carlo methods in the context of SDEs and we are interested in the cost for a fixed tolerance ϵ on the root mean square error. We describe theoretical bounds for the Monte Carlo cost rates in terms of the tolerance and we discuss under which conditions the MLMC method gives a better cost rate when compared to StMC. Also, we explain in detail how the parameters of the algorithm need to be chosen for a particular error tolerance.

3.2.1 Standard Monte Carlo

Let us introduce the general form of the Monte Carlo method. Suppose that $X(T) \in \mathbb{R}$ is the solution of a SDE at time T and consider a one-dimensional real-valued function $\phi : \mathbb{R} \to \mathbb{R}$. The function ϕ will be our quantity of interest (see also Section 2.4) and we set $\mathcal{P} = \phi(X(T))$. Next, we choose a particular number of timesteps $M = M_L = M_0 2^L$ (where $M_0, L \in \mathbb{N}$) in our discretisation. The corresponding timestep size is $T/M = h = h_L = \frac{T}{M_0 2^L}$. Choosing N_L independent samples, the StMC approximaton of $\mathbb{E}[\mathcal{P}]$ is given by

$$\hat{\mathcal{P}}_L^{(\text{StMC})} = \frac{1}{N_L} \sum_{i=1}^{N_L} \mathcal{P}_L^{(i)}. \tag{3.17}$$

In this expression $\mathcal{P}_L^{(i)} = \phi(X_{M_L}^{(i)})$ where $X_{M_L}^{(i)}$ is the approximation of the i^{th} independent sample path at time $T = M_L h_L$. The approximation $X_{M_L}^{(i)}$ can be obtained by using some timestepping method with M_L timesteps like for example those defined in Section 3.1. The timestep size h_L has the same use as the timestep h from Section 3.1 and L will later refer to the number of levels of the MLMC method. The size of h_L and the number of independent samples N_L depend on the required accuracy of the approximation as we see at the end of Section 3.2.2. For higher dimensional models the StMC estimator can be defined in the same way as equation (3.17).

Error and cost rates

We measure the error of the Monte Carlo approximation in terms of the mean square error which can be expressed as

$$\mathbb{E}\left[\left(\hat{\mathcal{P}}_L^{(\text{StMC})} - \mathbb{E}[\mathcal{P}]\right)^2\right] = \underbrace{\left(\mathbb{E}[\mathcal{P}_L] - \mathbb{E}[\mathcal{P}]\right)^2}_{\text{(squared) discretisation error}} + \underbrace{\frac{1}{N_L} Var[\mathcal{P}_L]}_{\text{(squared) sampling error}}, \tag{3.18}$$

where $\mathbb{E}[\mathcal{P}_L] = \mathbb{E}\left[\hat{\mathcal{P}}_L^{(\text{StMC})}\right] = \mathbb{E}\left[\mathcal{P}_L^{(i)}\right]$ and $Var[\mathcal{P}_L] = Var\left[\mathcal{P}_L^{(i)}\right]$ is the variance of $\mathcal{P}_L$. The first error on the right hand side of equation (3.18) is a result of the SDE discretisation and the second error arises due to Monte Carlo sampling for a finite number of samples N_L. For the mean square error to be bounded by a fixed tolerance ϵ^2 we bound both the discretisation and sampling errors by $\frac{\epsilon^2}{2}$.

The discretisation error depends on the numerical method that is used to approximate the sample path of $X(t)$. For the timestepping methods described in Section 3.1 we have shown numerically that the discretisation error is of order h_L and in Section 4.3.2 we show that (under some conditions) linear convergence can be proven theoretically. This means that

$$\mathbb{E}\left[\mathcal{P}_L\right] - \mathbb{E}\left[\mathcal{P}\right] = \mathcal{O}(h_L), \tag{3.19}$$

and therefore $h_L \propto \frac{\epsilon}{\sqrt{2}}$ which implies that the number of timesteps $M_L = T/h_L \propto \epsilon^{-1}$.

From the sampling error bound and since we require $\frac{1}{N_L} Var[\mathcal{P}_L] < \frac{\epsilon^2}{2}$ we have that the number of samples $N_L \propto \epsilon^{-2}$ and therefore the cost for a fixed error tolerance ϵ is given by

$$Cost^{(\text{StMC})} = \mathcal{O}(M_L \cdot N_L) \rightarrow Cost^{(\text{StMC})} = C^{(\text{StMC})}\epsilon^{-3} + \ldots, \tag{3.20}$$

which increases very fast as ϵ decreases. For example, if we want to increase the accuracy in the root mean square error by one decimal place (ie 10 times) then the cost increases 1000 times. As we will see later the MLMC method can improve the asymptotic cost rate to ϵ^{-2}. This improved rate is the best that can be achieved for a Monte Carlo method and this can be proved with the Central Limit Theorem. For a definition of the Central Limit Theorem you can see Lord et al. (2014).

As a last remark, we note that the overall cost, i.e. the constant $C^{(\text{StMC})}$ in (3.20) (but not the power of ϵ in (3.20)), also depends on the timestepping method and we examine this effect in our numerical results.

3.2.2 Multilevel Monte Carlo

We now define the MLMC method as in Giles (2008). The key idea for the MLMC approximation is that $\mathbb{E}\left[\mathcal{P}_L\right] = \mathbb{E}\left[\hat{\mathcal{P}}_L\right]$ can be written as the following telescoping sum

$$\mathbb{E}\left[\mathcal{P}_L\right] = \sum_{l=1}^{L} \mathbb{E}\left[\mathcal{P}_l - \mathcal{P}_{l-1}\right] + \mathbb{E}\left[\mathcal{P}_0\right]. \tag{3.21}$$

The MLMC method is then defined by

$$\hat{\mathcal{P}}_L^{(\text{MLMC})} = \sum_{l=0}^{L} \hat{Y}_{l,N_l},$$

where

$$\hat{Y}_{0,N_0} = \hat{\mathcal{P}}_0^{(\text{StMC})} \quad \text{and} \quad \hat{Y}_{l,N_l} = \frac{1}{N_l}\sum_{i=1}^{N_l} Y_l^{(i)}, Y_l^{(i)} = \mathcal{P}_l^{(i)} - \mathcal{P}_{l-1}^{(i)} \quad \text{for} \quad l \geq 1.$$

On the coarsest level ($l = 0$) we compute $\hat{Y}_{0,N_0}$ using the StMC method. This computation is generally cheap since the timestep size $h_0 = \frac{T}{M_0}$ is large. To correct the large discretisation error on the coarsest level, we compute corrections $\hat{Y}_{l,N_l}$ on the finer levels ($l > 0$) again by using StMC but this time on the difference $\mathcal{P}_l - \mathcal{P}_{l-1}$ with timestep sizes as described in Section 3.2.1.

For each $l \geq 0$ independent random variables are used to estimate $\hat{Y}_{l,N_l}$ so these estimates are independent for all l. However, on each pair of subsequent levels $l, l-1 \geq 0$ the same Brownian motion is used to estimate $\mathcal{P}_l$ and $\mathcal{P}_{l-1}$ in the estimator $\hat{Y}_{l,N_l}$. This guarantees that the variance of $\hat{Y}_{l,N_l}$ is small, and we will see below that this is crucial for the MLMC method to work. To couple the random variables on subsequent levels for a finite timestep size, a sum of normal random variables can be used in the simplest case (Symplectic Euler timestepping). In the following we give more details of how we couple the random variables for each timestepping method.

One of the main advantages of MLMC is that under certain conditions the cost is concentrated on the coarse levels (where the timestep size is larger) which are cheaper to compute since less timesteps are required. Also, it estimates differences between two estimators which have a smaller variance and therefore require a smaller number of samples. It is important to note that the method does not introduce any additional bias since

$$\mathbb{E}\left[\hat{\mathcal{P}}_L^{(\text{MLMC})}\right] = \sum_{l=1}^{L} \mathbb{E}\left[\mathcal{P}_l - \mathcal{P}_{l-1}\right] + \mathbb{E}\left[\mathcal{P}_0\right] = \mathbb{E}\left[\mathcal{P}_L\right] = \mathbb{E}\left[\hat{\mathcal{P}}_L^{(\text{StMC})}\right]. \tag{3.22}$$

Error and Cost rates:

In order to see how this method can reduce the computational cost we include a simplified version of the complexity theorem proved in Giles (2008). In Giles (2008) the computational cost is defined to be the total number of timesteps.

Theorem 3.1. *With the MLMC setting as above if there exist positive constants* c_1, c_2, c_3 *such that*

(i) $\left|\mathbb{E}[\hat{\mathcal{P}}_l - \mathcal{P}]\right| \leq c_1 h_l,$

(ii) $\mathbb{E}[\hat{Y}_{l,N_l}] = \begin{cases} \mathbb{E}[\hat{\mathcal{P}}_0] & \text{if} \quad l = 0, \\ \mathbb{E}[\hat{\mathcal{P}}_l - \hat{\mathcal{P}}_{l-1}] & \text{if} \quad l > 0, \end{cases}$

(iii) $Var[\hat{Y}_{l,N_l}] \leq c_2 N_l^{-1} h_l^2,$

(iv) C_l, *the computational complexity of* $\hat{Y}_{l,N_l}$, *is bounded by* $c_3 N_l h_l^{-1}$,

then there exists a positive constant c_4 *such that* $\forall \epsilon < e^{-1}$ *there are values* L *and* N_l *for which* $\hat{\mathcal{P}}_L^{(MLMC)}$ *satisfies*

$$\mathbb{E}\left[\left(\hat{\mathcal{P}}_L^{(MLMC)} - \mathbb{E}\left[\mathcal{P}\right]\right)^2\right] < \epsilon^2,$$

with the total number of timesteps bounded by

$$N_{steps} \leq c_4 \epsilon^{-2}. \tag{3.23}$$

Since the cost per timestep is constant, this implies that the total computational cost (i.e. the runtime of the algorithm) is given for fixed error tolerance ϵ *as*

$$Cost^{(MLMC)} = C^{(MLMC)} \epsilon^{-2} + \ldots, \tag{3.24}$$

where the constant $C^{(MLMC)}$ *depends on the particular timestepping method.*

Asymptotically the best exponent of ϵ you can get with a Monte Carlo method is -2. To improve this exponent further, Quasi Monte Carlo is required (Glasserman, 2003), (Giles and Waterhouse, 2009), (Kuo et al., 2015). As we have already seen, for the StMC method $Cost^{(\text{StMC})} = C^{(\text{StMC})} \epsilon^{-3} + \ldots$ and therefore we can see the importance of the MLMC method in the improvement of the efficiency of a numerical approximation.

Conditions (i) and (iv) are easily satisfied and they are true for the timestepping methods considered in this thesis. Also condition (ii) follows trivially from the definition of the MLMC method. However, condition (iii) is the hardest to achieve and requires the correct coupling of random variables on subsequent levels. In practice, we generate two independent normal random variables ξ_n and ξ_{n+1} for two consecutive

fine steps and then we couple them for a single coarse step. For the Symplectic Euler method this is achieved by the simple sum of normal random variables

$$\xi_n^{(\text{coarse})} = \frac{1}{\sqrt{2}}(\xi_n + \xi_{n+1}). \tag{3.25}$$

In Müller et al. (2015) it is shown that for Geometric Langevin the random variables are coupled using

$$\xi_n^{(\text{coarse})} = \frac{e^{-\lambda(X_n)h}\xi_n + \xi_{n+1}}{\sqrt{e^{-2\lambda(X_n)h} + 1}}, \tag{3.26}$$

and following exactly the same technique we couple the random variables of BAOAB as in equation (3.26) since the two methods have the same Ornstein-Uhlenbeck process.

In some cases it is possible to prove theoretically that Theorem 3.1 holds using either the theory of modified equations (Shardlow, 2006), (Zygalakis, 2011), (Müller et al., 2015) or strong convergence theory as we show in Sections 5.3 and 5.1. However, where the necessary assumptions are not satisfied, we verify that $Var[Y_l] \propto h_l^2$ numerically by plotting $Var[Y_l]$ as a function of h_l in a log-log plot, as is customary in the MLMC literature.

Finally, the mean square error of the MLMC method can be expressed as

$$\mathbb{E}\left[\left(\hat{\mathcal{P}}_L^{(\text{MLMC})} - \mathbb{E}\left[\mathcal{P}\right]\right)^2\right] = \underbrace{\left(\mathbb{E}\left[\mathcal{P}_L\right] - \mathbb{E}\left[\mathcal{P}\right]\right)^2}_{\text{(squared) discretisation error}} + \underbrace{\sum_{l=0}^{L} \frac{1}{N_l} Var[Y_{l,N_l}]}_{\text{(squared) sampling error}}. \tag{3.27}$$

We now describe how we pick the values of M_0, L and N_l on each level such that the sum of the (squared) discretisation and sampling errors is bounded by ϵ^2. For this, we first choose M_0 and L such that the bias error is bounded by $\frac{\epsilon}{\sqrt{2}}$. Note that this error only depends on $M_L = M_0 2^L$, so reducing the number of levels L by one and multiplying M_0 by two, leaves this error invariant (and in the StMC method there is freedom to pick any M_0, L as long as M_L is sufficiently large). In the MLMC method the number of levels L has to be carefully adjusted (while keeping M_L fixed) to reduce the cost from Monte Carlo sampling, as will be described below. In a second step, we then use a formula from Giles (2008) (see equation (3.30) below or equation (12) in Giles (2008)) to pick the optimal number of samples N_l on each level to minimise the cost, given a fixed tolerance $\epsilon/\sqrt{2}$ on the sampling error. This calculation of N_l is done

on-the-fly in the sampling algorithm.

To bound the discretisation error we first choose the values of M_0, T and then estimate the number of levels L as follows. From condition (i) of Theorem 3.1 it will be enough if $c_1 h_L \leq \frac{\epsilon}{\sqrt{2}}$, which can be written as

$$c_1 \frac{T}{M_0 2^L} \leq \frac{\epsilon}{\sqrt{2}}. \tag{3.28}$$

Then, to approximate the constant c_1 we assume that $\mathbb{E}[\hat{\mathcal{P}}_l - \mathcal{P}] = \tilde{c}_1 h_l$, for some $\tilde{c}_1 \in \mathbb{R}$ and we then write

$$\mathbb{E}[\hat{\mathcal{P}}_l - \hat{\mathcal{P}}_{l-1}] = \mathbb{E}[\hat{\mathcal{P}}_l - \mathcal{P}] - \mathbb{E}[\hat{\mathcal{P}}_{l-1} - \mathcal{P}] = \tilde{c}_1 h_l - \tilde{c}_1 h_{l-1} = \tilde{c}_1 h_l (1-2) = -\tilde{c}_1 h_l. \tag{3.29}$$

By approximating the left hand side numerically we find $c_1 = |\tilde{c}_1|$ and by using (3.28) we finally compute L. This method is described in Müller et al. (2015) and the same h_L is used for both StMC and MLMC methods since they have the same discretisation errors. As we see in Chapter 6, the splitting methods have smaller values of c_1 when compared to the Symplectic Euler method and this property affects the efficiency of the methods.

From above, the number of timesteps on the finest level equals $M_L = 2^L M_0$ and guarantees that the discretisation error is bounded by $\frac{\epsilon^2}{2}$. The value of M_0 however, which is the number of timesteps on the coarsest level, affects the efficiency of the MLMC method. For its choice we make sure that $Var[Y_1]$ is less than $\frac{1}{2} Var[Y_0]$. If this is not true then it is more efficient to increase the value of M_0 and drop some levels. For Symplectic Euler however, we need to be more careful and ensure that h_0 satisfies the stability constraint discussed in Section 3.1.

To bound the sampling error we need a special choice for the number of samples N_l and this is given in Giles (2008) by the following equation

$$N_l = \left\lceil 2\epsilon^{-2} \sqrt{Var(Y_l) h_l} \left(\sum_{i=0}^{L} \sqrt{Var(Y_i)/h_i} \right) \right\rceil. \tag{3.30}$$

This choice of number of samples on each level minimises the total cost and also ensures that the sampling error is bounded by $\frac{\epsilon^2}{2}$. The number of samples are then computed using on-the-fly estimators for $Var[Y_l]$. Practically we estimate the value of

$Var[Y_l]$ using an initial small number of samples and then compute N_l using equation 3.30. If the value of N_l for some l is larger than the initial number of samples we update the estimates for $Var[Y_l]$ for all l using the corresponding values of N_l and then compute new values for N_l for all l. This procedure continues until no more samples are needed.

Chapter 4

One-dimensional Model Analysis

Next we present some theoretical analysis for the one-dimensional dispersion models. Firstly, we prove the existence and uniqueness of a solution for the inhomogeneous model based on the existence of a Lyapunov function and the theory developed in Khasminskii (2011). Next, we discuss an alternative approach for the existence and uniqueness of solutions which is based on time change and the work in Wang and Zhang (2016). We continue by showing that our solution has bounded moments and then present the convergence results from Milstein and Tretyakov (2005) for timestepping methods applied to SDEs with nonglobally Lipschitz continuous coefficients. Finally, we show numerically that under some conditions it is possible to have explosion in the numerical approximation.

The application of the above results to atmospheric dispersion modelling are some of the main contributions of this thesis. To the best of our knowledge there was no previous application of these results in atmospheric dispersion.

For convenience, let us begin by summarising our two SDE models. Firstly, the homogeneous model is given by

$$\begin{aligned} dU(t) &= -\frac{U(t)}{\tau}dt + \sqrt{\frac{2\sigma_U^2}{\tau}}dW(t), \\ dX(t) &= U(t)dt, \end{aligned}$$

subject to deterministic initial conditions $U(0) = u_0$, $X(0) = x_0$ and with boundary conditions on $[0, H]$ given by $U(t^{\text{refl}} + h) = -U(t^{\text{refl}} - h)$ as $h \downarrow 0$ for t^{refl} such that $X(t^{\text{refl}}) = 0$ or $X(t^{\text{refl}}) = H$. Also, in this model τ and σ_U are constants.

Secondly, the inhomogeneous model is given by

$$dU(t) = \left[-\frac{U(t)}{\tau(X(t))} + \frac{1}{2}\left[1 + \left(\frac{U(t)}{\sigma_U(X(t))} \right)^2 \right] \frac{\partial \sigma_U^2(X(t))}{\partial X} \right] dt + \sqrt{\frac{2\sigma_U^2(X(t))}{\tau(X(t))}} dW(t),$$
$$dX(t) = U(t)dt,$$

subject to the same initial and boundary conditions as the homogeneous model and with the functions

$$\sigma_U(X) = \kappa_\sigma u^* \left(1 - \frac{X}{H} \right)^{\frac{3}{4}}, \quad \tau(X) = \kappa_\tau \frac{X}{\sigma_U(X)}, \tag{4.1}$$

for some constants $\kappa_\sigma, \kappa_\tau, u^*$ and H. Also note that near the boundaries we apply the Met Office's constant regularisation given by

$$\tau(X) = \tau(\epsilon_{\text{reg}}), \quad \sigma_U(X) = \sigma_U(\epsilon_{\text{reg}}) \text{ if } X < \epsilon_{\text{reg}}, \tag{4.2}$$

$$\tau(X) = \tau(H - \epsilon_{\text{reg}}), \quad \sigma_U(X) = \sigma_U(H - \epsilon_{\text{reg}}) \text{ if } X > H - \epsilon_{\text{reg}}, \tag{4.3}$$

for some small regularisation constant $0 < \epsilon_{\text{reg}} \ll H$. Finally, we also use the notation

$$\lambda(X) = \frac{1}{\tau(X)}, \quad \sigma(X) = \sqrt{\frac{2\sigma_U^2(X)}{\tau(X)}}, \tag{4.4}$$

$$\frac{\partial V}{\partial X}(X, U) = -\frac{1}{2}\left[1 + \left(\frac{U}{\sigma_U(X)} \right)^2 \right] \frac{\partial \sigma_U^2}{\partial X}(X). \tag{4.5}$$

4.1 Existence and uniqueness of solutions

We now show the existence and uniqueness of a non-explosive solution (explosion happens when the solution can be infinite in a finite time). In the literature (Øksendal, 2003), (Karatzas and Shreve, 2012) the most common proof for the existence and uniqueness of a solution requires that the drift term (the term in front of dt) and the diffusion term (the term in front of $dW(t)$) are both globally Lipschitz continuous. As it can be seen from the inhomogeneous model equations redefined above, the drift term contains the term U^2 which is clearly not Lipschitz continuous. Therefore, we need

to look for more relaxed conditions and study, for example, the work in Watanabe and Ikeda (1981) and Khasminskii (2011).

Our proof is for the inhomogeneous model since the drift term in the homogeneous model is linear in $U(t)$ and the diffusion coefficient is constant. These conditions are enough for the existence of a unique solution (Øksendal, 2003), (Karatzas and Shreve, 2012). In particular, in the absence of boundary conditions the homogeneous model is a simple Ornstein-Uhlenbeck process for which we can compute the exact solution (see Appendix A.2.1).

Since the model has boundary conditions we take them into account in the proof. Without loss of generality, let us set the boundary layer height $H = 1$. For our special choice of $\sigma_U(x)$ and $\tau(x)$ as in equations (4.1) (with a suitable regularisation as discussed later) we show that the solution does not explode in the set

$$S = \{(x, u) \in [0, 1] \times \mathbb{R}\}. \tag{4.6}$$

From the theory in Khasminskii (2011) it suffices to find a Lyapunov function with certain properties. In Lemma 4.2 we define the Lyapunov function that we are going to use in our proof but before that we give the following definition (Khasminskii, 2011).

Definition 4.1. *The infinitesimal generator $\tilde{\mathcal{L}}$ related to the inhomogeneous model equations* (2.7) *and* (2.8) *is defined by*

$$\tilde{\mathcal{L}}F = \left[-\frac{u}{\tau(x)} + \frac{1}{2}\left(1 + \frac{u^2}{\sigma_U^2(x)}\right)\frac{\partial \sigma_U^2(x)}{\partial x}\right]\frac{\partial F}{\partial u} + u\frac{\partial F}{\partial x} + \frac{\sigma_U^2(x)}{\tau(x)}\frac{\partial^2 F}{\partial u^2}, \tag{4.7}$$

where $F(x, u)$ is a twice differentiable function.

Next, we define our Lyapunov function in the following lemma.

Lemma 4.2. *If $\tau(x)$ and $\sigma_U^2(x)$ are smooth enough then the function*

$$F(x, u) = 1 + \frac{1}{2}\frac{u^2}{\sigma_U^2(x)} - \frac{1}{2}\log\frac{\sigma_U^2(x)}{(\kappa_\sigma u^*)^2}, \tag{4.8}$$

satisfies

$$\tilde{\mathcal{L}}F(x, u) \leq \frac{c}{\tau(x)}F(x, u) \qquad \forall (x, u) \in S, \tag{4.9}$$

where c is some positive constant and $\tilde{\mathcal{L}}$ is the infinitesimal generator of equations (2.7) *and* (2.8)*. We call the function F a Lyapunov function.*

$Proof$. Differentiating $F(x,u)$ with respect to x and u gives

$$\frac{\partial F}{\partial x} = -\frac{1}{2}\frac{u^2}{\sigma_U^4(x)}\frac{\partial \sigma_U^2(x)}{\partial x} - \frac{1}{2}\frac{1}{\sigma_U^2(x)}\frac{\partial \sigma_U^2(x)}{\partial x}, \quad \frac{\partial F}{\partial u} = \frac{u}{\sigma_U^2(x)}, \quad \frac{\partial^2 F}{\partial u^2} = \frac{1}{\sigma_U^2(x)}, \tag{4.10}$$

and substituting in equation (4.7) we obtain

$$\tilde{\mathcal{L}}F = \frac{1}{\tau(x)}\left(1 - \frac{u^2}{\sigma_U^2(x)}\right). \tag{4.11}$$

Combining with equation (4.8) we get

$$\tilde{\mathcal{L}}F = \frac{1}{\tau(x)}\left(3 - 2F - \log\frac{\sigma_U^2(x)}{(\kappa_\sigma u^*)^2}\right). \tag{4.12}$$

From equation (4.8) and the fact that $\frac{\sigma_U^2(x)}{(\kappa_\sigma u^*)^2} < 1$ for our choice of $\sigma_U(x)$, we then have that

$$1 \leq 1 + \frac{1}{2}\frac{u^2}{\sigma_U^2(x)} - \frac{1}{2}\log\frac{\sigma_U^2(x)}{(\kappa_\sigma u^*)^2} = F.$$

In addition,

$$-\log\frac{\sigma_U^2(x)}{(\kappa_\sigma u^*)^2} = 2F - 2 - \frac{u^2}{\sigma_U^2(x)} \leq 2F,$$

and therefore using equation (4.12) we obtain

$$\tilde{\mathcal{L}}F \leq \frac{3}{\tau(x)}F.$$

□

Equation (4.8) and Lemma 4.2 allow us to apply the theory from Khasminskii (2011) to show existence and uniqueness. The proof in Khasminskii (2011) is about a problem without boundary conditions so it is necessary to make some small changes which we indicate in the proof. Due to the boundary conditions, we define the set

$$S^{(n)} = \{(x,u) \in \mathbb{R}^2 : x \in \left[\frac{1}{n}, 1 - \frac{1}{n}\right], |u| \leq n\} \text{ for } n > 0.$$

(Note that if we didn't have boundary conditions we would use the set

$$S^{(n)} = \{(x,u) \in \mathbb{R}^2 : |(x,u)| \leq n\} \text{ for } n > 0,$$

as in Khasminskii (2011).)

The theory in Khasminskii (2011), adapted to our inhomogeneous model, begins by constructing a sequence of functions $a_1^{(n)}(x,u), a_2^{(n)}(x,u)$ and $\sigma^{(n)}(x,u)$ defined on the set S with the following properties:

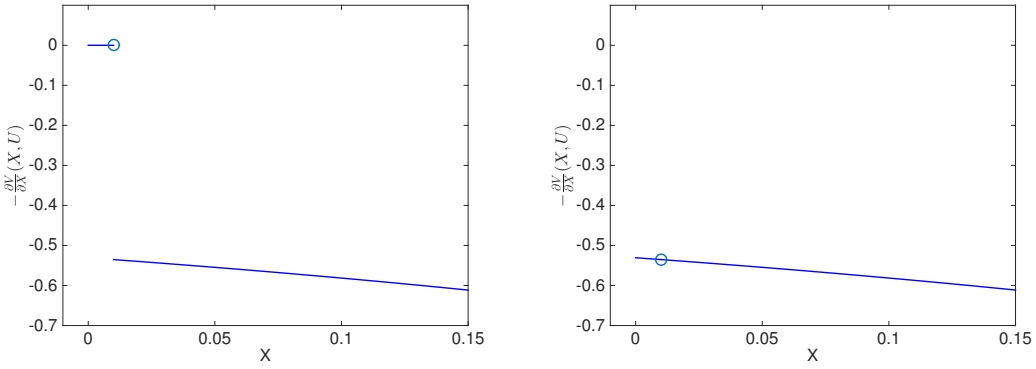
(i) The functions $a_1^{(n)}(x,u), a_2^{(n)}(x,u)$ and $\sigma^{(n)}(x,u)$ agree on the set $S^{(n)}$ with the drift and diffusion terms of equations (2.7) and (2.8), i.e.
$a_1^{(n)}(x,u) = -\lambda(x)u - \frac{\partial V(x,u)}{\partial x}, \quad a_2^{(n)}(x,u) = u, \quad \sigma^{(n)}(x,u) = \sigma(x)$ for all $(x,u) \in S^{(n)}$.

(ii) The functions $a_1^{(n)}(x,u), a_2^{(n)}(x,u)$ and $\sigma^{(n)}(x,u)$ are Lipschitz continuous and satisfy linear growth on the set S, i.e.
$\sum_{i=1}^{2} \left| a_i^{(n)}(x,u) - a_i^{(n)}(\tilde{x},\tilde{u}) \right| + \left| \sigma^{(n)}(x,u) - \sigma^{(n)}(\tilde{x},\tilde{u}) \right| \leq B \left| (x,u) - (\tilde{x},\tilde{u}) \right|,$
$\sum_{i=1}^{2} \left| a_i^{(n)}(x,u) \right| + \left| \sigma^{(n)}(x,u) \right| \leq B(1 + |(x,u)|),$
for all $(x,u),(\tilde{x},\tilde{u}) \in S$ and some positive constant $B > 0$. $|\cdot|$ denotes the Euclidean norm.

Let us now discuss the above two properties for our inhomogeneous model. Property (i) follows trivially since we just set the new sequence to be equal to the original SDE coefficients on the set $S^{(n)}$. Property (ii) will follow by extending the new sequence outside the set $S^{(n)}$ and also if the original SDE coefficients are continuous on $S^{(n)}$. Continuity however is not true for our model since the Met Office's constant regularisation (equations (2.24) and (2.25)) introduces a discontinuity in the drift term. In particular it introduces a discontinuity in the potential term $\frac{\partial V}{\partial X}(X,U)$ at the points $X = \epsilon_{\text{reg}}$ and $X = H - \epsilon_{\text{reg}}$ (see beginning of Chapter 4) and as a result the function $a_1^{(n)}(x,u)$ cannot be Lipschitz continuous. In Figure 4-1a we plot the potential function for values of X near the ground and we can see the discontinuity at the point $X = \epsilon_{\text{reg}}$.

To avoid introducing this discontinuity, we replace the regularisation introduced in (2.24) and (2.25) by the following smooth regularisation that leads to no discontinuities in the parameter functions at ϵ_{reg}:

$$\tau(X) = \kappa_\tau \frac{(X^2 + \epsilon_{\text{reg}}^2)^{\frac{1}{2}}}{\sigma_U(X)}, \quad \sigma_U(X) = \kappa_\sigma u^* \left(\frac{\left((H-X)^2 + \epsilon_{\text{reg}}^2\right)^{\frac{1}{2}}}{H} \right)^{\frac{3}{4}}. \tag{4.13}$$



(a) Constant regularisation

(b) Smooth regularisation

Figure 4-1: $\frac{\partial V}{\partial X}(X,U)$ plots using a constant and a smooth regularisarion. The circle indicates the value at the point $X = \epsilon_{\text{reg}}$.

In Figure 4-1b we plot the smoothed potential function and we can see that there are no discontinuities. In Chapter 5 we will see a different type of a smooth regularisation that has the extra advantage of removing the new discontinuities introduced by the boundary conditions treatment described in Section 5.2.

With the above assumptions, the newly constructed sequence of functions now satisfy Lipschitz continuity and linear growth on the set S so the system of SDE's

$$dU^{(n)}(t) = a_1^{(n)}(X^{(n)}(t), U^{(n)}(t))dt + \sigma^{(n)}(X^{(n)}(t), U^{(n)}(t))dW(t), \tag{4.14}$$

$$dX^{(n)}(t) = a_2^{(n)}(X^{(n)}(t), U^{(n)}(t))dt, \tag{4.15}$$

has an almost surely unique solution $(X^{(n)}(t), U^{(n)}(t))$ on S (Øksendal, 2003), (Karatzas and Shreve, 2012).

Now note that for $m \geq n$ the first exit time of $(X^{(m)}(t), U^{(m)}(t))$ from $S^{(n)}$ equals the first exit time of $(X^{(n)}(t), U^{(n)}(t))$ from $S^{(n)}$ so we use ρ_n to denote the common value of that exit time. Also, the two processes $(X^{(n)}(t), U^{(n)}(t))$ and $(X^{(m)}(t), U^{(m)}(t))$ coincide up to time ρ_n. Then, define

$$\rho = \lim_{n\to\infty} \rho_n, \tag{4.16}$$

which is the first exit time from every set $S^{(n)}$. Lastly, we need the following definition.

Definition 4.3. *The process $(X(t), U(t))$ is called regular (or non explosive) if for all*

$(x, u) \in S$ *we have that*

$$\mathbb{P}\left(\rho < \infty, |U_\rho| = \infty | (X(0), U(0)) = (x, u)\right) = 0, \tag{4.17}$$

where ρ *is defined by equation* (4.16)*.*

In Khasminskii (2011) only the condition $\rho < \infty$ is included in Definition 4.3 because they study a problem without boundary conditions. If equation (4.17) is true then $(X(t), U(t))$ is defined almost surely for all $t \geq 0$.

In Theorem 4.5 (Khasminskii, 2011) we show that under an assumption on $\tau(x)$ the inhomogeneous model has a unique regular solution.

Assumption 4.4. *The function* $\tau(x)$ *is bounded below by a positive constant, i.e. there exists a constant* $\tilde{B} > 0$ *such that*

$$\tau(x) \geq \tilde{B} \quad \forall x \in [0, 1]. \tag{4.18}$$

Equation (4.18) is true when we use the Met Office's (equations (2.24) and (2.25)) or the smooth regularisation as described above (equation (4.13)).

Theorem 4.5. *If* $\tau(x)$ *is bounded below by a positive constant and if there exist a nonnegative, twice differentiable function* $F(x, u)$*, defined on* S *such that for some constant* $\tilde{c} > 0$ *we have*

$$\mathcal{L}F(x, u) \leq \tilde{c}F(x, u), \tag{4.19}$$

$$F_n := \inf_{|u|>n} F(x, u) \to \infty \quad \textit{as} \quad n \to \infty, \tag{4.20}$$

where $\mathcal{L} = \frac{\partial}{\partial t} + \tilde{\mathcal{L}}$*, then the inhomogeneous model with initial conditions on* $S^{(n)}$ *for every* n*, has a unique regular solution up to the first reflection time.*

$Proof$. Using Lemma 4.2 and the fact that $F(x, u)$ does not depend on t we know that under Assumption 4.4 the nonnegative function

$$F(x, u) = 1 + \frac{1}{2}\frac{u^2}{\sigma_U^2(x)} - \frac{1}{2}\log\frac{\sigma_U^2(x)}{(\kappa_\sigma u^*)^2}, \tag{4.21}$$

satisfies condition (4.19) and it also satisfies condition (4.20). The proof is then identical to the one of Theorem 3.5 in Khasminskii (2011) and here we only give a sketch proof.

Define the function

$$W(t,x,u) = F(x,u)\exp(-\tilde{c}t), \tag{4.22}$$

which satisfies $\mathcal{L}W \leq 0$ by using condition (4.19). Then, by Lemma 3.2 in Khasminskii (2011) (based on Itô's formula applied to $W(t,x,u)$) we have

$$\begin{aligned}&\mathbb{E}\left[F(X(\rho_n \wedge t), U(\rho_n \wedge t))\exp(-\tilde{c}(\rho_n \wedge t))\right] - \mathbb{E}\left[F(X(0),U(0))\right] \\ &= \mathbb{E}\left[\int_0^{\rho_n \wedge t} \mathcal{L}W(s,X(s),U(s))ds\right] \leq 0,\end{aligned} \tag{4.23}$$

where $\rho_n \wedge t = min\{\rho_n, t\}$ with ρ_n the exit time as above.

Now define a process $(\tilde{X}(t), \tilde{U}(t)) = (X^{(n)}(t), U^{(n)}(t))$ for $t < \rho_n$. Using inequality (4.23) with the fact that $\rho_n \wedge t \leq t$ we have

$$\mathbb{E}\left[F(\tilde{X}(\rho_n \wedge t), \tilde{U}(\rho_n \wedge t))\right] \leq \exp(\tilde{c}t)\mathbb{E}\left[F(X(0),U(0))\right]. \tag{4.24}$$

Also,

$$\begin{aligned}\mathbb{E}\left[F(\tilde{X}(\rho_n \wedge t), \tilde{U}(\rho_n \wedge t))\right] &\geq \mathbb{E}\left[F(\tilde{X}(\rho_n \wedge t), \tilde{U}(\rho_n \wedge t))\mathbb{1}_{\{\rho_n \leq t, |U(\rho_n)|=n\}}\right] \\ &= \mathbb{E}\left[F(\tilde{X}(\rho_n), \tilde{U}(\rho_n))\mathbb{1}_{\{\rho_n \leq t, |U(\rho_n)|=n\}}\right] \\ &\geq \inf_{x\in[0,1],|u|\geq n}\{F(x,u)\}\mathbb{E}\left[\mathbb{1}_{\{\rho_n \leq t, |U(\rho_n)|=n\}}\right] \\ &= \inf_{x\in[0,1],|u|\geq n}\{F(x,u)\}\mathbb{P}\left[\rho_n \leq t, |U(\rho_n)| = n\right],\end{aligned}$$

and combining with inequality (4.24) we get

$$\mathbb{P}\left[\rho_n \leq t, |U(\rho_n)| = n\right] \leq \frac{\exp(\tilde{c}t)\mathbb{E}\left[F(X(0),U(0))\right]}{\inf_{x\in[0,1],|u|\geq n}\{F(x,u)\}}. \tag{4.25}$$

By letting $n \to \infty$ and using condition (4.20) gives equation (4.17), so the process $(\tilde{X}(t), \tilde{U}(t))$ and therefore $(X(t), U(t))$ is an almost surely unique regular solution to the inhomogeneous model up to time ρ.

Since equation (4.17) holds we know that the exit time ρ corresponds to the time when the particle hits the boundary of the set S and therefore the proof is complete.

□

Next, in order to extend our solution to an arbitrary time t, we need to show that the solution does not explode when the particle has initial conditions on the boundary of the set S. Since the boundary of S lies outside $S^{(n)}$ for every n, Theorem 4.5 cannot be applied in this case.

An easy method to solve this issue would be the application of a special type of regularisation for $\tau(X)$ and $\sigma_U(X)$ as we do for example in Section 5.3.2 where our SDE coefficients become continuously differentiable. If this regularisation guarantees that the solution does not explode before the particle enters some set $S^{(n)}$ then our solution exists for an arbitrary time t. For example, if we assume that $\tau(X)$ and $\sigma_U(X)$ are constants near the boundary (as the Met Office does) then our model becomes a simple Ornstein-Uhlenbeck process for which we know that a solution exists. Therefore, with this simple approach we can extend our solution to an arbitrary time t.

4.1.1 An alternative approach

In this section we discuss an alternative approach to prove existence and uniqueness of solutions that uses time change and works for a class of models with a singularity. Also, we will use material from Wang and Zhang (2016) where the authors study SDEs with non-Lipschitz coefficients. Our arguments will not be completely rigorous and making a rigorous proof could be a topic for future work.

Firstly, we give some definitions from Wang and Zhang (2016).

Definition 4.6. *An increasing function $\phi : \mathbb{R}^+ \to \mathbb{R}^+$ is called a Dini function if*

$$\int_0^1 \frac{\phi(t)}{t} dt < \infty. \tag{4.26}$$

Definition 4.7. *A function f on the Euclidean space is called Hölder-Dini continuous of order $\alpha \in [0, 1)$ if*

$$|f(x) - f(y)| \leq |x - y|^\alpha \phi(|x - y|), \quad |x - y| \leq 1, \tag{4.27}$$

holds for some Dini function ϕ and is called Dini continuous if this condition holds for $\alpha = 0$.

We then have the following result from Wang and Zhang (2016). SDEs with locally Dini continuous drift term (the term in front of dt) and locally Hölder-Dini continuous diffusion term (the term in front of $dW(t)$) of order $\frac{2}{3}$, have a unique strong solution.

Before applying this result to our inhomogeneous model equations (2.7) and (2.8) we will firstly apply a time change technique to our equations which removes the singularity at $X = 0$ (we consider only this singularity since we do a sketch proof). Define

$$\tilde{t} = \int_0^t \frac{1}{\tau(X(s))} ds, \quad d\tilde{t} = \frac{1}{\tau(X(t))} dt, \tag{4.28}$$

to obtain

$$\begin{aligned} dU(\tilde{t}) = &-U(\tilde{t})d\tilde{t} - \frac{1}{2}\tau(X(\tilde{t}))\left[1 + \frac{U^2(\tilde{t})}{\sigma_U^2(X(\tilde{t}))}\right]\frac{\partial \sigma_U^2}{\partial X}(X(\tilde{t}))d\tilde{t} \\ &+ \sqrt{2\tau(X(\tilde{t}))}\sigma_U(X(\tilde{t}))dW(\tilde{t}), \end{aligned} \tag{4.29}$$

$$dX(\tilde{t}) = U(\tilde{t})\tau(X(\tilde{t}))d\tilde{t}, \tag{4.30}$$

by substituting (4.28) in (2.7) and (2.8) and also using the chain rule for the diffusion term.

The time change in (4.28) however is not well-defined if we use our form of $\tau(X)$ since it contains the term $1/X$, but it does allow singularities of the form $1/X^\alpha$ for $\alpha < 1$. To make the time change well-defined in our case, one method could be to replace $1/X$ by $1/X^{1-\epsilon}$, for some small ϵ.

Our drift terms now become

$$\begin{aligned} -U - \frac{1}{2}\tau(X)\left[1 + \frac{U^2}{\sigma_U^2(X)}\right]\frac{\partial \sigma_U^2}{\partial X}(X) = &-U - \frac{3}{4H}\kappa_\tau \kappa_\sigma u^* X \left[\frac{1}{(1-\frac{X}{H})^{\frac{1}{4}}}\right. \\ &\left. + \frac{U^2}{(\kappa_\sigma u^*)^2(1-\frac{X}{H})^{\frac{7}{4}}}\right], \end{aligned} \tag{4.31}$$

$$U\tau(X) = \frac{\kappa_\tau}{\kappa_\sigma u^*}\frac{UX}{(1-\frac{X}{H})^{\frac{3}{4}}}, \tag{4.32}$$

and our diffusion term becomes

$$\sqrt{2\tau(X)}\sigma_U(X) = \sqrt{2\kappa_\tau \kappa_\sigma u^*}\sqrt{X(1-\frac{X}{H})^{\frac{3}{4}}}. \tag{4.33}$$

We can now easily see that this method removes the singularity at $X = 0$. Then, by assuming some suitable regularisation we make our new SDE coefficients (4.31), (4.32) and (4.33) locally Lipschitz continuous.

Now, if the drift terms are locally Lipschitz continuous they are also locally Dini continuous with Dini function a constant multiple of the identity function. Also, if the diffusion term is locally Lipschitz continuous then it is also locally Hölder-Dini continuous of order $\frac{2}{3}$ with Dini function a constant multiple of the identity function raised to the power of $\frac{1}{3}$. Therefore, using the theory from Wang and Zhang (2016) we have that when the particle hits the boundary at $X = 0$ the solution does not explode before the particle enters any set $S^{(n)}$. Theorem 4.5 can then be apllied until the next time the particle hits the boundary and the existence-uniqueness proof is now complete.

The advantage of this approach when compared to the proof of theorem 4.5 is that it removes the singularity at $X = 0$ and also does not require to consider separately the case when the particle starts from the boundary. However, further work is needed to make this method rigorous and this could be a topic for future work.

4.2 Bounded moments for the SDE solution

As a consequence of the existence of the Lyapunov function (equation (4.8)) we can also show that the random variable U_t has bounded moments. For the random variable X_t a similar result is not required since the particles are restricted in the interval $[0, 1]$. In this proof, we make the following assumption:

Assumption 4.8. *The function $\sigma_U^2(x)$ is bounded above by a positive constant, i.e there exists a constant $\tilde{b} > 0$ such that*

$$\sigma_U^2(x) \leq \tilde{b} \quad \forall x \in [0, 1]. \tag{4.34}$$

Lemma 4.9. *If $\sigma_U^2(x)$ is bounded above by some positive constant then the solution U_t of the inhomogeneous model has a bounded second moment.*

$Proof$. Define the set

$$\tilde{S}^{(n)} = \{(x,u) \in \mathbb{R}^2 : x \in [0,1], |u| \leq n\} \text{ for } n > 0, \tag{4.35}$$

and let $\tilde{\rho}_n$ be the first exit time of (X_t, U_t) from $\tilde{S}^{(n)}$ (note that $\tilde{S}^{(n)}$, $\tilde{\rho}_n$ are different from $S^{(n)}$, ρ_n used in Section 4.1). Then, as in equations (4.23) and (4.24) we have

$$\begin{aligned}\mathbb{E}\left[F(X(\tilde{\rho}_n \wedge t), U(\tilde{\rho}_n \wedge t))\right] &\leq \exp(\tilde{c}t)\mathbb{E}\left[F(X(0), U(0))\right] \\ &= \exp(\tilde{c}t)F(X(0), U(0)).\end{aligned} \tag{4.36}$$

where $\tilde{\rho}_n \wedge t = min\{\tilde{\rho}_n, t\}$. Substituting equation (4.8) gives

$$\mathbb{E}\left[1 + \frac{1}{2}\frac{U^2(\tilde{\rho}_n \wedge t)}{\sigma_U^2(X(\tilde{\rho}_n \wedge t))} - \frac{1}{2}\log\frac{\sigma_U^2(X(\tilde{\rho}_n \wedge t))}{(\kappa_\sigma u^*)^2}\right] \leq \exp(\tilde{c}t)F(X(0), U(0)), \tag{4.37}$$

and using linearity of expectation we obtain

$$\mathbb{E}\left[\frac{U^2(\tilde{\rho}_n \wedge t)}{\sigma_U^2(X(\tilde{\rho}_n \wedge t))}\right] \leq 2\exp(\tilde{c}t)F(X(0), U(0)) + \mathbb{E}\left[\log\frac{\sigma_U^2(X(\tilde{\rho}_n \wedge t))}{(\kappa_\sigma u^*)^2}\right] - 2. \tag{4.38}$$

Then, by using Assumption 4.8, equation (4.38) becomes

$$\mathbb{E}\left[U^2(\tilde{\rho}_n \wedge t)\right] \leq 2\tilde{b}\exp(\tilde{c}t)F(X(0), U(0)) + \tilde{b}\log\tilde{b} - \tilde{b}\log(\kappa_\sigma u^*)^2 - 2\tilde{b}. \tag{4.39}$$

Finally, letting $n \to \infty$ gives $\tilde{\rho}_n \to \infty$ (since $U(t)$ does not explode) and since $t \leq T$ we obtain

$$\mathbb{E}\left[U^2(t)\right] \leq 2\tilde{b}\exp(\tilde{c}T)F(X(0), U(0)) + \tilde{b}\log\tilde{b} - \tilde{b}\log(\kappa_\sigma u^*)^2 - 2\tilde{b}. \tag{4.40}$$

The bound on the right hand side depends only on the deterministic initial conditions which completes the proof.

□

To obtain bounds for higher moments of $U(t)$ we can apply Itô's formula to higher powers of $\exp(-\tilde{c}t)F(X(t), U(t))$ and follow a similar method as in Lemma 4.9.

4.3 Convergence of the timestepping methods

In this section we present some convergence results for the timestepping methods that we use to discretise our SDE models. In order to describe in which sense our approximations converge we consider the general SDE

$$d\mathbf{X}(t) = f(\mathbf{X}(t))dt + G(\mathbf{X}(t))d\mathbf{W}(t), \tag{4.41}$$

with $f : \mathbb{R}^d \to \mathbb{R}^d, G : \mathbb{R}^d \to \mathbb{R}^{d\times m}$ and $\mathbf{W}(t)$ an m-dimensional Brownian motion (see Definition 2.1). Also, we need a special class of functions $\phi : \mathbb{R}^d \to \mathbb{R}$ given in the following definition from Lord et al. (2014).

Definition 4.10. *The function $\phi : \mathbb{R}^d \to \mathbb{R}$ is said to have a polynomially bounded r-derivative if there exists $p \in \mathbb{N}$ such that*

$$\sup_{\mathbf{x}\in\mathbb{R}^d} \left\{ \left| \frac{\partial^{|r|}\phi}{\partial x_1^{r_1}\partial x_2^{r_2}\cdots\partial x_d^{r_d}}(\mathbf{x}) \right| / \left(1 + \|\mathbf{x}\|_2^p\right) \right\} < \infty, \tag{4.42}$$

where $|r| = r_1 + r_2 + \ldots + r_d$ and $\|\cdot\|_2$ denotes the Euclidean norm.

In the next definition we describe how we measure the approximation errors of the above general SDE.

Definition 4.11. *(a) For a sufficiently small timestep size h, we have that the approximation $\mathbf{X}_M$ of the exact solution $\mathbf{X}(T)$ obtained using some timestepping method (see Section 3.1) and $M = T/h$ number of timesteps is said to have a weak order α error if for some constant $C > 0$*

$$|\mathbb{E}\left[\phi(\mathbf{X}(T))\right] - \mathbb{E}\left[\phi(\mathbf{X}_M)\right]| \leq Ch^\alpha, \tag{4.43}$$

where $\phi : \mathbb{R}^d \to \mathbb{R}$ is any infinitely differentiable function with polynomially bounded derivatives.

(b) Also, we say that the strong error is of order α if

$$\mathbb{E}\left(|\mathbf{X}(T) - \mathbf{X}_M|\right) \leq \tilde{C}h^\alpha, \tag{4.44}$$

for some constant $\tilde{C} > 0$.

Note that for the Monte Carlo approximations we measure the errors in terms of the root mean square error defined in Section 3.2 (see equations (3.18) and (3.27)).

Most common results for the convergence of timestepping methods require a global Lipschitz condition (Lipschitz continuous on the whole domain) on the SDE coefficients like for example in property (ii) in the construction of the functions $a_1^{(n)}(x,u)$, $a_2^{(n)}(x,u)$ and $\sigma^{(n)}(x,u)$ before proving Theorem 4.5. Since our inhomogeneous model (equation (2.7) with a smooth regularisation) only satisfies a local Lipschitz condition (Lipschitz continuous on closed and bounded subsets of the domain) these methods cannot guarantee the convergence of the approximation.

In Kloeden and Platen (2011) they prove that Lipschitz continuity and linear growth of the SDE coefficients (property (ii) mentioned in the previous paragraph) imply that the strong error of the Euler-Maruyama method is of order $\frac{1}{2}$ (Theorem 10.2.2). If in addition the coefficients are sufficiently smooth (up to order 4) with polynomially bounded derivatives, then the weak error is of order 1 (Theorem 14.5.1). In Bou-Rabee and Owhadi (2010) the Geometric Langevin method is analysed and if $\lambda(x)$ and $\sigma(x)$ are constants and V depends only on $X(t)$ (see equation (2.7)), is Lipschitz continuous and satisfies a special regularity condition, then the strong error has order 1 (Theorem 2.1). Since the homogeneous model has linear coefficients, the convergence of the timestepping methods follows trivially and therefore we consider only the inhomogeneous model.

4.3.1 Weak convergence of timestepping methods applied to inhomogeneous model

To deal with the absence of globally Lipschitz continuous coefficients we look for more relaxed conditions and in particular study theoretical approaches that only require a local Lipschitz condition, as in Higham et al. (2002) and Milstein and Tretyakov (2005). We review some general results for a wider class of SDEs and then apply them to our inhomogeneous model.

One of the results in Higham et al. (2002) involves the proof of strong convergence for Euler-type timestepping methods applied to SDEs with locally Lipschitz continuous coefficients under the assumption that the exact and numerical solutions have bounded p moments for some $p > 2$. Consequently, in order to apply this result to our model the only additional requirement would be to prove that the numerical solution

has bounded moments (bounded moments of the exact solution follow from Section 4.2). However, this proof turned out to be challenging for our non-linear inhomogeneous model and it might also not be true, since a numerical experiment in Section 4.4 shows that under some conditions the approximation can explode.

Instead of showing that the numerical solution has bounded moments, we apply the theory from Milstein and Tretyakov (2005) which is based on the existence of a Lyapunov function (as the theory in Sections 4.1 and 4.2) and on rejecting trajectories that go outside some set $\tilde{S}^{(R)}$ where

$$\tilde{S}^{(R)} = \{(x,u) \in \mathbb{R}^2 : x \in [0,1], |u| < R\}, \tag{4.45}$$

with R a positive real constant (thus extending the definition of $\tilde{S}^{(n)}$ in (4.35)). (The sets differ from those in Milstein and Tretyakov (2005), i.e.

$$\tilde{S}^{(R)} = \{(x,u) \in \mathbb{R}^2 : |(x,u)| < R\}, \tag{4.46}$$

since that paper considers a problem without boundary conditions.)

The first requirement is that the SDE coefficients are sufficiently smooth and this can easily be obtained for our model in equation (2.7) by using the smooth regularisation described by the equations in (4.13). The second requirement is the existence of a Lyapunov function satisfying the conditions described by equations (4.19) and (4.20) which is also valid in our case by using equation (4.8) (see Lemma 4.2). The only difference between our theory and what is used in Milstein and Tretyakov (2005) is that in equation (4.20) we have $|u| > n$ instead of $|(x,u)| > n$ since we know that $X(t)$ is bounded due to the reflective boundary conditions. However, this difference does not affect the final result and the proof stays the same.

A key step in Milstein and Tretyakov (2005) is to define a special transformation on the initial random variables that will transform the given SDE to an equation with Lipschitz continuous and bounded coefficients. This is achieved with the function $\psi : \mathbb{R} \to \mathbb{R}$ defined by

$$\psi(u) = \begin{cases} u & \text{if} \quad -R' \le u \le R', \\ R' + \int_{R'}^{u} \frac{du'}{1+(u'-R')^k} & \text{if} \quad u > R', \\ -R' - \int_{u}^{-R'} \frac{du'}{1+(-R'-u')^k} & \text{if} \quad u < -R', \end{cases} \tag{4.47}$$

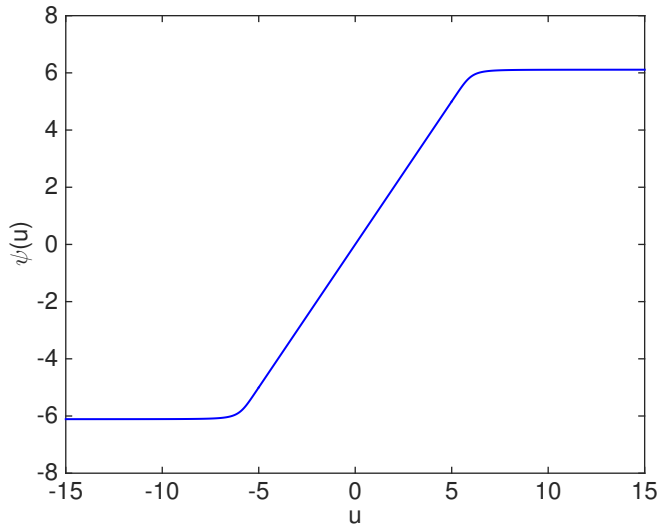


Figure 4-2: The transformation function $\psi(u)$ with $R' = 5$ and $k = 4$.

where $k \geq 2$ is an integer and $R' = R + r$ for some constant $r > 0$. The function ψ has bounded derivatives on $\mathbb{R}$ up to order k. As a consequence, for any function $g(x, u)$ that has continuous derivatives up to order k we have that the function

$$g^R(x, u) = g(\psi(x), \psi(u)), \tag{4.48}$$

has continuous and bounded derivatives up to order k. Since in our case we know that $x \in [0, 1]$ we will use

$$g^R(x, u) = g(x, \psi(u)). \tag{4.49}$$

Also, since the function ψ is continuous and bounded there exists a constant $\rho > r$, independent of R, such that $(x, \psi(u)) \in \tilde{S}^{(R+\rho)}$ for every $(x, u) \in [0, 1] \times \mathbb{R}$. A plot of the transformation function ψ with $R' = 5$ and $k = 4$ can be seen in Figure 4-2.

By setting $X^R(t) = X(t)$, $U^R(t) = \psi(U(t))$, $X^R(0) = X(0)$ and $U^R(0) = U(0)$, where $(X(t), U(t))$ is the solution of the inhomogeneous model, we can construct the following system of SDEs (Milstein and Tretyakov, 2005)

$$dU^R(t) = -\lambda(X^R(t))U^R(t)dt - \frac{\partial V^R}{\partial X}(X^R(t), U^R(t))dt + \sigma(X^R(t))dW(t), \tag{4.50}$$

$$dX^R(t) = U^R(t)dt, \tag{4.51}$$

whose coefficients have bounded derivatives up to order k. The next proposition is then proved in Milstein and Tretyakov (2005).

Proposition 4.12. *Assume that $F(x,u)$ satisfies* (4.19)*,* (4.20) *and*

$$\frac{\min\limits_{x\in[0,1],|u|\geq R+\rho} F(x,u)}{\min\limits_{x\in[0,1],|u|\geq R} F(x,u)} \leq \alpha, \tag{4.52}$$

where α is a constant independent of R. Let $\phi(x,u)$ be a function such that

$$\lim_{R\to\infty} \frac{\max\limits_{x\in[0,1],|u|\leq R} |\phi(x,u)|}{\min\limits_{x\in[0,1],|u|\geq R} F(x,u)} = 0. \tag{4.53}$$

Then for all $(X(0),U(0))\in[0,1]\times\mathbb{R}$ and all $\epsilon>0$ there exists $R(X(0),U(0),\epsilon)>0$ such that $\forall R > R(X(0),U(0),\epsilon)$

$$\left|\mathbb{E}\left[\phi^R(X^R(T),U^R(T))\right] - \mathbb{E}\left[\phi(X(T),U(T))\right]\right| < \epsilon. \tag{4.54}$$

The significance of Proposition 4.12 is that we can construct a new regularised model with Lipschitz continuous coefficients by introducing an arbitrarily small extra error. With some simple algebra (see Lemma 4.13) we can easily see that this proposition applies to our inhomogeneous problem since equation (4.8) and the quantities of interest defined in Section 2.4 satisfy the required conditions. The result from Proposition 4.12 will also be used later in Chapter 5 in the application of Theorem 3.1.

Lemma 4.13. *The function*

$$F(x,u) = 1 + \frac{1}{2}\frac{u^2}{\sigma_U^2(x)} - \frac{1}{2}\log\frac{\sigma_U^2(x)}{(\kappa_\sigma u^*)^2}, \tag{4.55}$$

and the quantities of interest defined in Section 2.4 satisfy the conditions of Proposition 4.12 with $R\geq 1$.

Proof. We have already seen in Lemma 4.2 and Theorem 4.5 that $F(x,u)$ satisfies

(4.19) and (4.20). For inequality (4.52), we have

$$\begin{aligned}\frac{\min\limits_{x\in[0,1],|u|\geq R+\rho} F(x,u)}{\min\limits_{x\in[0,1],|u|\geq R} F(x,u)} &= \frac{\min\limits_{x\in[0,1],|u|\geq R+\rho}\left\{1+\frac{1}{2}\frac{u^2}{\sigma_U^2(x)}-\frac{1}{2}\log\frac{\sigma_U^2(x)}{(\kappa_\sigma u^*)^2}\right\}}{\min\limits_{x\in[0,1],|u|\geq R}\left\{1+\frac{1}{2}\frac{u^2}{\sigma_U^2(x)}-\frac{1}{2}\log\frac{\sigma_U^2(x)}{(\kappa_\sigma u^*)^2}\right\}}\\ &= \frac{1+\frac{(R+\rho)^2}{\alpha_1}+\alpha_2}{1+\frac{R^2}{\alpha_1}+\alpha_2},\end{aligned}$$

where $\alpha_1 = \max\limits_{x\in[0,1]} \sigma_U^2(x)$ and $\alpha_2 = \min\limits_{x\in[0,1]}\left\{-\frac{1}{2}\log\frac{\sigma_U^2(x)}{(\kappa_\sigma u^*)^2}\right\}$. Then,

$$\begin{aligned}\frac{\min\limits_{x\in[0,1],|u|\geq R+\rho} F(x,u)}{\min\limits_{x\in[0,1],|u|\geq R} F(x,u)} &\leq \frac{1+\frac{2(R^2+\rho^2)}{\alpha_1}+\alpha_2}{1+\frac{R^2}{\alpha_1}+\alpha_2} \leq \frac{1+\frac{2(R^2+\rho^2)}{\alpha_1}+\alpha_2}{\frac{R^2}{\alpha_1}}\\ &= \frac{\alpha_1}{R^2}+2\left(1+\frac{\rho^2}{R^2}\right)+\frac{\alpha_1\alpha_2}{R^2}\\ &\leq \alpha_1+2\left(1+\rho^2\right)+\alpha_1\alpha_2 \quad (\text{since } R\geq 1).\end{aligned}$$

The right hand side is independent of R so by setting it equal to α we obtain inequality (4.52).
From Section 2.4 we have that $\phi(x,u)$ equals $X(T)$ or $\mathbb{1}_{[a,b]}(X(T))$ and therefore

$$\max_{x\in[0,1],|u|\leq R} |\phi(x,u)| = 1. \tag{4.56}$$

Substituting in equation (4.53) gives

$$\lim_{R\to\infty} \frac{\max\limits_{x\in[0,1],|u|\leq R}|\phi(x,u)|}{\min\limits_{x\in[0,1],|u|\geq R} F(x,u)} = \lim_{R\to\infty} \frac{1}{\min\limits_{x\in[0,1],|u|\geq R} F(x,u)} = 0, \tag{4.57}$$

by property (4.20).

□

Finally, the following two assumptions on the timestepping method (in our case Symplectic Euler, Geometric Langevin and BAOAB) used to discretise the Lipschitz continuous system of equations (4.50) and (4.51) are necessary before stating the main convergence theorem.

Assumption 4.14. *The timestepping method applied to the regularised SDEs* (4.50)

and (4.51) *has a weak error (see Definition 4.11) of at least order 1 and after one timestep the local weak error (i.e. Definition 4.11 after one timestep) is of at least order 2.*

Assumption 4.15. *If* $(X_n^R, U_n^R) \in \tilde{S}^{(R)}$ *for* $n = 0, \ldots, k$*, where* (X_n^R, U_n^R) *is the approximation of the regularised SDEs* (4.50) *and* (4.51) *at time* t_n*, then* $(X_n, U_n) = (X_n^R, U_n^R)$ *for* $n = 0, \ldots, k$*.*

And now we state the main convergence theorem from Milstein and Tretyakov (2005).

Theorem 4.16. *Consider any timestepping method satisfying Assumption 4.14 which converges with a weak error of order* p *for systems with sufficiently smooth and bounded derivatives up to some order (e.g. order* 4 *for Euler-Maruyama). Let the conditions of Proposition 4.12 and the Assumption 4.15 be fulfilled and consider a function* $\phi(x, u)$ *such that* $\mathbb{E}\left[|\phi(X(T), U(T))|\right]$ *exists. Then for all* $(X(0), U(0)) \in [0, 1] \times \mathbb{R}$ *and* $\epsilon > 0$ *there exists* $R(X(0), U(0), \epsilon) > 0$ *such that* $\forall R \geq R(X(0), U(0), \epsilon)$ *and sufficiently small* h

$$\left|\mathbb{E}\left[\phi(X(T), U(T))\right] - \mathbb{E}\left[\phi(X_M, U_M)\mathbb{1}_{\Omega_R}(\omega)\right]\right| < Kh^p + \epsilon, \tag{4.58}$$

where $\Omega_R = \{\omega : (X_n^R, U_n^R) \in \tilde{S}^{(R)}, n = 0, \ldots, M-1$ *and* $(X_M^R, U_M^R) \in [0, 1] \times [-R, R]\}$ *and* $K > 0$ *depends on* $(X(0), U(0))$ *and* R*.*

The only change we did in Theorem 4.16 is to assume that $\mathbb{E}\left[|\phi(X(T), U(T))|\right]$ exists (true for our quantities of interest since the mean particle position and the concentration are bounded) instead of assuming that $|\phi(x, u)| \leq \tilde{V}(x, u)$. In Milstein and Tretyakov (2005) the second assumption is used to prove the first assumption but if we assume directly that $\mathbb{E}\left[|\phi(X(T), U(T))|\right]$ exists then Theorem 4.16 holds for a wider class of functions $\phi(x, u)$.

The conclusion from Theorem 4.16 is that by rejecting the trajectories of particles that have a velocity with absolute value greater than R we can guarantee the weak convergence of the timestepping method by just adding a small error. For example, when we use the Euler-Maruyama method, we have that equation (4.58) holds with $p = 1$. Note that increasing R decreases the extra error ϵ but at the same time it can also increase K so h needs to be sufficiently small.

We already know that for the regularised SDEs (4.50) and (4.51), Assumption 4.14 holds for the Euler-Maruyama method (Kloeden and Platen, 2011). In Section 4.3.2

we show that if Assumption 4.14 is true for the Euler-Maruyama method then it is also true for all the timestepping methods defined in Section 3.1. Also, Assumption 4.15 trivially holds for our timestepping methods since the coefficients of the original SDEs and the Lipschitz continuous system in equations (4.50) and (4.51) are equal on the set $\tilde{S}^{(R)}$.

4.3.2 A Taylor series expansion approach for weak convergence

With a Taylor series expansion approach we show that if the Euler-Maruyama method converges with a weak error of order 1 then the Symplectic Euler, Geometric Langevin and BAOAB methods also converge with at least the same order of weak error. Here we only present the analysis for the Symplectic Euler method but a similar approach works for Geometric Langevin and BAOAB (see Appendix A.3 for the Geometric Langevin method). Consequently, the convergence result from Milstein and Tretyakov (2005) (see Theorem 4.16) will also hold for the timestepping methods described in Section 3.1 when applied to the inhomogeneous model.

For our analysis we use the following theorem from Kloeden and Platen (2011) for the weak convergence of a general timestepping method.

Theorem 4.17. *Using the notation introduced at the beginning of Section 4.3, a method converges with weak error of order α if the local error over one timestep satisfies*

$$|\mathbb{E}\left[\phi(\mathbf{X}(h))\right] - \mathbb{E}\left[\phi(\mathbf{X}_1)\right]| \leq Ch^{\alpha+1}, \tag{4.59}$$

for all polynomials ϕ up to degree $2\alpha + 1$ and a constant $C > 0$.

For example, for the Euler-Maruyama method, Theorem 4.17 is true with $\alpha = 1$ (see beginning of Section 4.3). Using Taylor series expansions we show that this is also true for the Symplectic Euler, Geometric Langevin and BAOAB methods.

Firstly, let $(X_1^{\text{SE}}, U_1^{\text{SE}})$ be the Symplectic Euler approximation to the exact solution $(X(h), U(h))$ of the inhomogeneous model after one timestep as described by equations (3.1) and (3.2). Denote by $(X_1^{\text{EM}}, U_1^{\text{EM}})$ the corresponding approximation for the Euler - Maruyama method. Our result for the Symplectic Euler method is then summarised in the following theorem.

Corollary 4.18. *If the Euler-Maruyama method applied to SDEs* (2.7) *and* (2.8) *converges with a weak error of order* 1 *then the weak error of the Symplectic Euler method,*

applied to the same equations, also converges with at least order 1.

$Proof$. Let ϕ be a polynomial as in Theorem 4.17. For the Symplectic Euler method we have

$$\begin{aligned}\left|\mathbb{E}\left[\phi(X(h),U(h))\right]-\mathbb{E}\left[\phi(X_1^{\text{SE}},U_1^{\text{SE}})\right]\right| = &\left|\mathbb{E}\left[\phi(X(h),U(h))\right]\right. \\ &\left.-\mathbb{E}\left[\phi(X_0+U_1^{\text{SE}}h,U_1^{\text{SE}})\right]\right|, \qquad (4.60)\end{aligned}$$

and since the only difference that Euler-Maruyama has, is that it uses $X_1^{\text{EM}}=X_0+U_0h$ instead of $X_1^{\text{SE}}=X_0+U_1^{\text{SE}}h$ we apply a Taylor series expansion on the first argument of $\phi(X_0+U_1^{\text{SE}}h,U_1^{\text{SE}})$. This gives

$$\begin{aligned}\phi(X_0+U_1^{\text{SE}}h,U_1^{\text{SE}}) = &\ \phi(X_0+[(1-\lambda(X_0)h)\,U_0-\frac{\partial V}{\partial X}(X_0,U_0)h+\sigma(X_0)\sqrt{h}\xi_0]h,U_1^{\text{SE}}) \\ = &\ \phi(X_0+U_0h-\lambda(X_0)U_0h^2-\frac{\partial V}{\partial X}(X_0,U_0)h^2+\sigma(X_0)h^{\frac{3}{2}}\xi_0,U_1^{\text{SE}}) \\ = &\ \phi(X_0+U_0h,U_1^{\text{SE}}) \\ &+\frac{\partial\phi}{\partial X}(X_0+U_0h,U_1^{\text{SE}})(-\lambda(X_0)U_0h^2-\frac{\partial V}{\partial X}(X_0,U_0)h^2+\sigma(X_0)h^{\frac{3}{2}}\xi_0) \\ &+\text{higher order terms.}\end{aligned}$$

Since we are interested in expectations we can neglect all higher order terms since they either have a mean zero or they are of order higher than h^2. Substituting in (4.60) and using the fact that $U_1^{\text{SE}}=U_1^{\text{EM}}$, we obtain

$$\begin{aligned}\left|\mathbb{E}\left[\phi(X(h),U(h))\right]-\mathbb{E}\left[\phi(X_1^{\text{SE}},U_1^{\text{SE}})\right]\right| &= \left|\mathbb{E}\left[\phi(X(h),U(h))\right]\right. \\ &\quad\left.-\mathbb{E}\left[\phi(X_0+U_0h,U_1^{\text{SE}})\right]+\mathcal{O}(h^2)\right| \\ &\le \left|\mathbb{E}\left[\phi(X(h),U(h))\right]\right. \\ &\quad\left.-\mathbb{E}\left[\phi(X_0+U_0h,U_1^{\text{SE}})\right]\right|+\mathcal{O}(h^2) \\ &= \left|\mathbb{E}\left[\phi(X(h),U(h))\right]\right. \\ &\quad\left.-\mathbb{E}\left[\phi(X_1^{\text{EM}},U_1^{\text{EM}})\right]\right|+\mathcal{O}(h^2).\end{aligned}$$

If the Euler-Maruyama method converges weakly with order 1 we have

$$\left|\mathbb{E}\left[\phi(X(h),U(h))\right]-\mathbb{E}\left[\phi(X_1^{\text{EM}},U_1^{\text{EM}})\right]\right|\le C^{\text{EM}}h^2, \qquad (4.61)$$

for some constant $C^{\text{EM}} > 0$ which implies

$$\left|\mathbb{E}\left[\phi(X(h), U(h))\right] - \mathbb{E}\left[\phi(X_1^{\text{SE}}, U_1^{\text{SE}})\right]\right| = \mathcal{O}(h^2), \tag{4.62}$$

and completes the proof by using Theorem 4.17.

□

The same approach as in Corollary 4.18 but with harder algebra also works for Geometric Langevin and BAOAB. See Appendix A.3 for the Geometric Langevin method.

Our conclusion is that for a system of SDEs with Lipschitz continuous, sufficiently smooth and polynomially bounded coefficients we have that our timestepping methods converge with a weak error of order 1 (since the Euler-Maruyama does (Kloeden and Platen, 2011)). As a result, by rejecting any exploding trajectories as described in Milstein and Tretyakov (2005) we can guarantee the convergence of our timestepping methods when applied to the inhomogeneous model. First order weak convergence will also be used in Section 5.3.2 in the derivation of the modified equations (Shardlow, 2006), (Zygalakis, 2011).

4.4 A simple numerical experiment with the Symplectic Euler method

Next, we present some simple numerical experiments to demonstrate that for our inhomogeneous model and values of $U(0)$ which are physically valid, it is very unlikely that the approximation of $U(t)$ becomes large. Also, to show that it is possible to have explosive behaviour in extreme cases, we present an experiment with an unphysically large value of $U(0)$. Although the explosive behaviour that we observe might not be related to the discussion of Section 4.3.1 and it is simply related to instability, i.e. the timestep size is not small enough for the given $U(0)$, we still present this experiment to show that it is unlikely to have any problems with our set of parameters.

We always use the Symplectic Euler method and the inhomogeneous model parameters $\kappa_\sigma = 1.3$, $\kappa_\tau = 0.5$, $u^* = 0.2$, $H = 1$, $\epsilon_{\text{reg}} = 0.01km$, $T = 1$ and $X(0) = 0.05$. The value of $U(0)$ depends on the experiment. Denote by $U_n^{(i)}$ the approximation of the i^{th} trajectory at time $t_n = nh$.

For our first experiment we compute the maximum and minimum values of U_n

on every timestep for 1000 trajectories. We use two different timestep sizes and a physically valid, large value of $U(0)$ which equals 10 in this case. For the timestep $h = 1/(40 \cdot 2^4) = 1.5625 \cdot 10^{-3}$ we obtain

$$\min_{1\leq i\leq 1000, 0\leq n\leq T/h} U_n^{(i)} = -6.56, \qquad \max_{1\leq i\leq 1000, 0\leq n\leq T/h} U_n^{(i)} = 10, \tag{4.63}$$

and for $h = 1/(40 \cdot 2^6) = 3.90625 \cdot 10^{-4}$ we obtain

$$\min_{1\leq i\leq 1000, 0\leq n\leq T/h} U_n^{(i)} = -7.32, \qquad \max_{1\leq i\leq 1000, 0\leq n\leq T/h} U_n^{(i)} = 10.01. \tag{4.64}$$

We observe that in both cases the absolute value of $U_n^{(i)}$ does not become large and is of the order of $U(0)$. Therefore, a relatively small value of R will be enough to guarantee weak convergence of the method for this choice of parameters.

In our numerical experiments in Chapter 6, we use a smaller value for $U(0)$ $(= 0.1)$ and as a result it is even more unlikely that U_n becomes large after time $T = 1$. No explosive behaviour is observed and therefore we do not check whether any trajectory goes outside some set $\tilde{S}^{(R)}$. Also, we assume that the methods converge with weak order 1 as suggested by the numerical experiments.

However, starting with a large value of $U(0)$ can lead to instability. This is indeed the case for $U(0) = 2000$ and $h = 1/(40 \cdot 2^4) = 1.5625 \cdot 10^{-3}$. As can be seen in Figure 4-3, this happens very soon and after just 4 timesteps. The main reason for this behaviour is that in the Euler step

$$U_{n+1} = \left(1 - \frac{1}{\tau(X_n)}h\right) U_n + \frac{1}{2}\left[1 + \left(\frac{U_n}{\sigma_U(X_n)}\right)^2\right] \frac{\partial \sigma_U^2}{\partial X}(X_n)h + \sqrt{\frac{2\sigma_U^2(X_n)}{\tau(X_n)}}\sqrt{h}\xi_n, \tag{4.65}$$

$$X_{n+1} = X_n + U_{n+1}h, \tag{4.66}$$

we compute U_{n+1} by taking the square of U_n and if the timestep h which multiplies U_n^2 is not small enough then the approximation becomes large very quickly. The value of U_n continues to increase and after some point, all the computational time is spent by the particles reflecting within the boundary layer.

By reducing the timestep size to $h = 1/(40 \cdot 2^6) = 3.90625 \cdot 10^{-4}$ results in no instability after time $T = 1$ (i.e. the approximation does not become large very

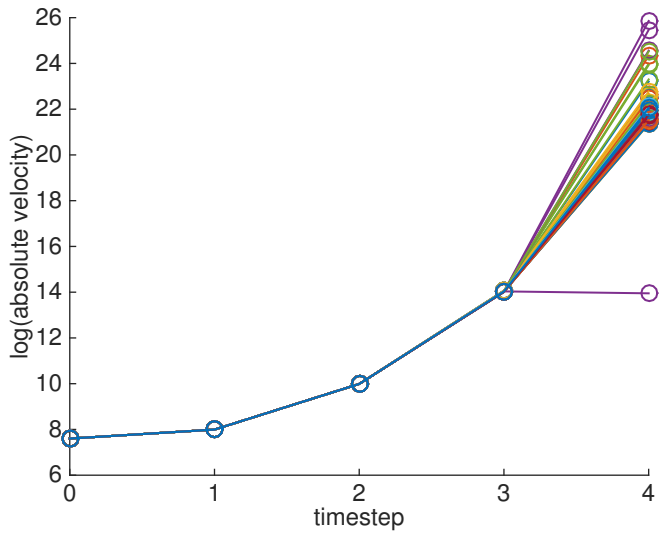


Figure 4-3: Explosive behaviour for large U_0 (= 2000)

quickly). However reducing the timestep size cannot guarantee a stable approximation since with the Gaussian random variables there is always a probability (although very small) that the random increment of equation (4.65) can lead to a large increase in the value of U_{n+1}. If this increase requires a smaller timestep size then we have instability again. Therefore, by rejecting any trajectories that exceed a certain value can guarantee a stable solution.

Our conclusion is that for small and physical values of $U(0)$ (< 10) it is very unlikely that any trajectories will have large values and the method from Milstein and Tretyakov (2005) has little effect on the results. In addition, in Chapter 6, with the given choice of model parameters and initial conditions we can see based on numerical results that it is reasonable to assume a first order weak convergence without checking for any exploding trajectories. For larger values of $U(0)$ however and if h is not small enough, we have evidence that the approximation could become large and therefore checking for explosive trajectories becomes very important.

Chapter 5

MLMC and the application of the Complexity Theorem

Our next topic is to study the theoretical application of Theorem 3.1 (complexity theorem, Giles (2008)) to the one-dimensional models that will enable us to determine the conditions under which the MLMC method gives a better asymptotic cost rate when compared to StMC.

We begin from the homogeneous model and present a direct approach for this application to a problem without reflection. We continue by presenting the theoretical analysis of a new treatment of reflective boundary conditions that preserves the quadratic MLMC variance decay which is one of the most important and hardest to show requirements of Theorem 3.1. With this new algorithm that verifies numerically Theorem 3.1, we then describe how a modified equations approach (Shardlow, 2006), (Zygalakis, 2011) (Müller et al., 2015) can be used to show the theoretical application to the inhomogeneous model.

Finally, we discuss how we deal with the discontinuity of the indicator function in the concentration problem. This discontinuity causes an increase in the variance which reduces the performance of the MLMC. Therefore, it becomes necessary to find a suitable approximation and our approach is to use smoothing poynomials (Giles et al., 2015), (Giles et al., 2017).

Some of these results have already been published in Katsiolides et al. (2018). To the best of our knowledge this is the first time that the theoretical application of Theorem 3.1 (complexity theorem, Giles (2008)) is studied for a problem from atmospheric

dispersion modelling.

5.1 Homogeneous model

Firstly, we prove that the conditions of Theorem 3.1 hold for the simplified model and the Geometric Langevin method without boundary conditions using a uniform timestep. For the problem with reflection, the application will follow with the special boundary conditions treatment described later in Section 5.2. Also, without boundary conditions the homogeneous model is a simple Ornstein-Uhlenbeck process for which a well-defined exact solution exists. We do not include any results for the other two timestepping methods since the simplified model is not the main focus of this thesis.

As a quantity of interest we take the mean particle position so in the notation used in Section 3.2 we have that $\mathcal{P} = X(T)$. We use the index (f) to denote the fine step approximations and the index (c) to denote the coarse step approximations. From the simplified model equations (2.13), (2.14) and the discretisation method equations (3.8), (3.9) and (3.10) we have that on level l and n even the fine step becomes

$$U_{n+1}^{(f)} = e^{-\lambda h_l} U_n^{(f)} + \sigma \alpha_{h_l} \xi_n, \tag{5.1}$$

$$X_{n+1}^{(f)} = X_n^{(f)} + U_{n+1}^{(f)} h_l, \tag{5.2}$$

and the coarse step

$$U_{n+2}^{(c)} = e^{-2\lambda h_l} U_n^{(c)} + \sigma \alpha_{2h_l} \xi_n^{\star}, \tag{5.3}$$

$$X_{n+2}^{(c)} = X_n^{(c)} + U_{n+2}^{(c)} 2h_l, \tag{5.4}$$

with $\alpha_h = \sqrt{\frac{1-e^{-2\lambda h}}{2\lambda}}$, $\xi_n \sim Normal(0,1)$ i.i.d. and $\xi_n^{\star} = \frac{r\xi_n + \xi_{n+1}}{\sqrt{r^2+1}}$. Also let $r = e^{-\lambda h_l}$. The fine and coarse steps have the same initial conditions so

$$X_0^{(f)} = X_0^{(c)}, \quad U_0^{(f)} = U_0^{(c)}. \tag{5.5}$$

Lemma 5.1. *If the product λh_l is small enough then conditions $(i) - (iv)$ of Theorem 3.1 hold for the simplified model and the Geometric Langevin method.*

$Proof$. The Geometric Langevin method, applied to the simplified model, is a first order method (Bou-Rabee and Owhadi, 2010) which implies that condition (i) fol-

lows. Clearly, condition (ii) follows from the definition of the MLMC method and the linearity of expectation. Also, we know that the cost on each level is proportional to the number of samples multiplied with the number of timesteps (since the cost per timestep is fixed) and therefore condition (iv) follows. It only remains to prove that condition (iii) holds, i.e.

$$Var[\hat{Y}_{l,N_l}] \leq CN_l^{-1}h_l^2,$$

for some positive constant C.
From equation (5.2),

$$X_{n+2}^{(f)} = X_n^{(f)} + \left(U_{n+1}^{(f)} + U_{n+2}^{(f)}\right) h_l, \tag{5.6}$$

which implies using equation (5.4)

$$X_{n+2}^{(f)} - X_{n+2}^{(c)} = X_n^{(f)} - X_n^{(c)} + \left(U_{n+1}^{(f)} + U_{n+2}^{(f)} - 2U_{n+2}^{(c)}\right) h_l. \tag{5.7}$$

From equation (5.1),

$$U_{n+2}^{(f)} = r\left(rU_n^{(f)} + \sigma\sqrt{\frac{1-r^2}{2\lambda}}\xi_n\right) + \sigma\sqrt{\frac{1-r^2}{2\lambda}}\xi_{n+1} \tag{5.8}$$

$$= r^2U_n^{(f)} + \sigma r\sqrt{\frac{1-r^2}{2\lambda}}\xi_n + \sigma\sqrt{\frac{1-r^2}{2\lambda}}\xi_{n+1}, \tag{5.9}$$

which implies using equation (5.3)

$$\begin{aligned}
U_{n+2}^{(f)} - U_{n+2}^{(c)} &= r^2U_n^{(f)} + \sigma\sqrt{\frac{1-r^2}{2\lambda}}\left(r\xi_n + \xi_{n+1}\right) - r^2U_n^{(c)} - \sigma\sqrt{\frac{1-r^4}{2\lambda}}\frac{r\xi_n + \xi_{n+1}}{\sqrt{r^2+1}} \\
&= r^2U_n^{(f)} + \sigma\sqrt{\frac{1-r^2}{2\lambda}}\left(r\xi_n + \xi_{n+1}\right) - r^2U_n^{(c)} - \sigma\sqrt{\frac{1-r^2}{2\lambda}}\left(r\xi_n + \xi_{n+1}\right) \\
&= r^2\left(U_n^{(f)} - U_n^{(c)}\right) \\
&= r^{n+2}\left(U_0^{(f)} - U_0^{(c)}\right) \qquad \text{(by induction)} \\
&= 0.
\end{aligned}$$

Therefore, equation (5.7) now becomes

$$\begin{aligned} X_{n+2}^{(f)} - X_{n+2}^{(c)} &= X_n^{(f)} - X_n^{(c)} + \left(U_{n+1}^{(f)} - U_{n+2}^{(c)}\right) h_l \\ &= X_0^{(f)} - X_0^{(c)} + h_l \sum_{m=0}^{\frac{n}{2}} \left(U_{n-2m+1}^{(f)} - U_{n-2m+2}^{(c)}\right) \qquad \text{(by induction)} \\ &= h_l \sum_{m=0}^{\frac{n}{2}} \left(U_{n-2m+1}^{(f)} - U_{n-2m+2}^{(c)}\right) \qquad (\text{since} X_0^{(f)} = X_0^{(c)}). \end{aligned} \tag{5.10}$$

In order to evaluate the above sum we firstly compute the difference $U_{n+1}^{(f)} - U_{n+2}^{(c)}$. From equation (5.1)

$$\begin{aligned} U_{n+1}^{(f)} &= r^2 U_{n-1}^{(f)} + \sigma\alpha_{h_l} \left(r\xi_{n-1} + \xi_n\right) \\ &= r^{n+1} U_0^{(f)} + \sigma\alpha_{h_l} \sum_{k=0}^{n} r^{n-k} \xi_k \\ &= r^{n+1} U_0^{(f)} + \sigma\alpha_{h_l} \sum_{k=0}^{\frac{n}{2}-1} \left(r^{n-2k}\xi_{2k} + r^{n-2k-1}\xi_{2k+1}\right) + \sigma\alpha_{h_l}\xi_n. \end{aligned}$$

From equation (5.3)

$$\begin{aligned} U_{n+2}^{(c)} &= r^4 U_{n-2}^{(c)} + \sigma\alpha_{2h_l} \left(r^2\xi_{n-2}^{\star} + \xi_n^{\star}\right) \\ &= r^{n+2} U_0^{(c)} + \sigma\alpha_{2h_l} \sum_{k=0}^{\frac{n}{2}} r^{2(\frac{n}{2}-k)} \xi_{2k}^{\star} \\ &= r^{n+2} U_0^{(c)} + \sigma\alpha_{2h_l} \sum_{k=0}^{\frac{n}{2}} r^{2(\frac{n}{2}-k)} \frac{r\xi_{2k} + \xi_{2k+1}}{\sqrt{r^2+1}} \\ &= r^{n+2} U_0^{(c)} + \sigma\sqrt{\frac{1-r^2}{2\lambda}} \sum_{k=0}^{\frac{n}{2}} r^{n-2k} \left(r\xi_{2k} + \xi_{2k+1}\right) \\ &= r^{n+2} U_0^{(c)} + \sigma\alpha_{h_l} \sum_{k=0}^{\frac{n}{2}} \left(r^{n-2k+1}\xi_{2k} + r^{n-2k}\xi_{2k+1}\right). \end{aligned}$$

Subtracting the last two equations for $U_{n+1}^{(f)}$ and $U_{n+2}^{(c)}$ we get

$$U_{n+1}^{(f)} - U_{n+2}^{(c)} = \; r^{n+1} U_0^{(f)} - r^{n+2} U_0^{(c)}$$

$$
\begin{aligned}
&+ \sigma\alpha_{h_l} \sum_{k=0}^{\frac{n}{2}-1} \left[\left(r^{n-2k} - r^{n-2k+1} \right) \xi_{2k} + \left(r^{n-2k-1} - r^{n-2k} \right) \xi_{2k+1} \right] \\
&+ \sigma\alpha_{h_l}\xi_n - \sigma\alpha_{h_l} r\xi_n - \sigma\alpha_{h_l}\xi_{n+1} \\
=\ & r^{n+1}U_0^{(f)} - r^{n+2}U_0^{(c)} \\
&+ \sigma\alpha_{h_l} \sum_{k=0}^{\frac{n}{2}-1} \left[r^{n-2k}(1-r)\xi_{2k} + r^{n-2k-1}(1-r)\xi_{2k+1} \right] \\
&+ \sigma\alpha_{h_l}(1-r)\xi_n - \sigma\alpha_{h_l}\xi_{n+1} \\
=\ & r^{n+1}U_0^{(f)} - r^{n+2}U_0^{(c)} \\
&+ \sigma\alpha_{h_l} \left[(1-r) \sum_{k=0}^{\frac{n}{2}-1} \left(r^{n-2k}\xi_{2k} + r^{n-2k-1}\xi_{2k+1} \right) + (1-r)\xi_n - \xi_{n+1} \right].
\end{aligned}
$$

Substituting in equation (5.10) now gives

$$
\begin{aligned}
X_{n+2}^{(f)} - X_{n+2}^{(c)} =\ & h_l \sum_{m=0}^{\frac{n}{2}} \Big[r^{n-2m+1}U_0^{(f)} - r^{n-2m+2}U_0^{(c)} \\
&+ \sigma\alpha_{h_l} \big[(1-r) \sum_{k=0}^{\frac{n}{2}-m-1} (r^{n-2m-2k}\xi_{2k} + r^{n-2m-2k-1}\xi_{2k+1}) \\
&+ (1-r)\xi_{n-2m} - \xi_{n-2m+1} \big] \Big],
\end{aligned}
$$

and taking the variance

$$
\begin{aligned}
Var\left(X_{n+2}^{(f)} - X_{n+2}^{(c)} \right) =\ & h_l^2\sigma^2\alpha_{h_l}^2 Var \left(\sum_{m=0}^{\frac{n}{2}} \left[(1-r) \sum_{k=0}^{\frac{n}{2}-m-1} (r^{n-2m-2k}\xi_{2k} + r^{n-2m-2k-1}\xi_{2k+1}) \right. \right. \\
&\left. \left. + (1-r)\xi_{n-2m} - \xi_{n-2m+1} \right] \right) \\
=\ & h_l^2\sigma^2\alpha_{h_l}^2 \Bigg[Var \left(\sum_{m=0}^{\frac{n}{2}} \left[(1-r) \sum_{k=0}^{\frac{n}{2}-m-1} r^{n-2m-2k}\xi_{2k} + (1-r)\xi_{n-2m} \right] \right) \\
&+ Var \left(\sum_{m=0}^{\frac{n}{2}} \left[(1-r) \sum_{k=0}^{\frac{n}{2}-m-1} r^{n-2m-2k-1}\xi_{2k+1} - \xi_{n-2m+1} \right] \right) \Bigg] \\
&\text{(by independence of even and odd indexed random variables)}
\end{aligned}
$$

$$= h_l^2\sigma^2\alpha_{h_l}^2\Bigg[Var\underbrace{\left(\sum_{m=0}^{\frac{n}{2}}\Big[(1-r)\sum_{k=0}^{\frac{n}{2}-m} r^{n-2m-2k}\xi_{2k}\Big]\right)}_{A}$$
$$+ Var\underbrace{\left(\sum_{m=0}^{\frac{n}{2}}\Big[(1-r)\sum_{k=0}^{\frac{n}{2}-m-1} r^{n-2m-2k-1}\xi_{2k+1} - \xi_{n-2m+1}\Big]\right)}_{B}\Bigg].$$

Let us firstly consider sum A which contains all the even indexed random variables. By the construction of the sum and for a fixed integer l we can write the coefficient of ξ_{2l} as $(1-r)f_{2l}(r)$ where $f_{2l}(r)$ is a polynomial of r of degree at most n and with coefficients $1, 0$ or -1. Similarly, for the sum B we can write the coefficient of ξ_{2l+1} as $(1-r)f_{2l+1}(r) - 1$. Therefore, $Var\left(X_{n+2}^{(f)} - X_{n+2}^{(c)}\right)$ can be written as

$$\begin{aligned}
Var\left(X_{n+2}^{(f)} - X_{n+2}^{(c)}\right) &= h_l^2\sigma^2\alpha_{h_l}^2\Bigg[Var\left((1-r)\sum_{l=0}^{\frac{n}{2}} f_{2l}(r)\xi_{2l}\right) \\
&\quad + Var\left(\sum_{l=0}^{\frac{n}{2}}\Big[(1-r)f_{2l+1}(r) - 1\Big]\xi_{2l+1}\right)\Bigg] \\
&= h_l^2\sigma^2\alpha_{h_l}^2\left[(1-r)^2\sum_{l=0}^{\frac{n}{2}}(f_{2l}(r))^2 + \sum_{l=0}^{\frac{n}{2}}\Big[(1-r)f_{2l+1}(r) - 1\Big]^2\right] \\
&\quad \text{(by independence)} \\
&\leq h_l^2\sigma^2\alpha_{h_l}^2\left[(1-r)^2\sum_{l=0}^{\frac{n}{2}}(f_{2l}(r))^2 + \sum_{l=0}^{\frac{n}{2}}\Big[2(1-r)^2(f_{2l+1}(r))^2 + 2\Big]\right].
\end{aligned}$$

By the triangle inequality we have that $|f_l(r)| \leq \sum_{k=0}^{n} r^k = \frac{1-r^{n+1}}{1-r}$ which implies that

$$\begin{aligned}
Var\left(X_{n+2}^{(f)} - X_{n+2}^{(c)}\right) &\leq h_l^2\sigma^2\alpha_{h_l}^2\left[(1-r)^2\sum_{l=0}^{\frac{n}{2}}\Big(\frac{1-r^{n+1}}{1-r}\Big)^2 + \sum_{l=0}^{\frac{n}{2}}\Big[2(1-r)^2\Big(\frac{1-r^{n+1}}{1-r}\Big)^2 + 2\Big]\right] \\
&= h_l^2\sigma^2\alpha_{h_l}^2\left(\frac{n}{2}+1\right)\left[(1-r^{n+1})^2 + 2(1-r^{n+1})^2 + 2\right].
\end{aligned} \tag{5.11}$$

Let $F(r) = (1 - r^{n+1})^2 + 2(1 - r^{n+1})^2 + 2$. We have that $0 \leq r = e^{-\lambda h_l} \leq 1$ which implies that $F(r) \leq \tilde{C}$ for some positive constant $\tilde{C}$. We can then write inequality (5.11) as

$$\begin{aligned} Var\left(X_{n+2}^{(f)} - X_{n+2}^{(c)}\right) &\leq h_l^2 \alpha_{h_l}^2 \sigma^2 \tilde{C} \left(\frac{n}{2} + 1\right) \\ &= \tilde{C}\sigma^2 h_l^2 \frac{n+2}{2} \frac{1 - e^{-2\lambda h_l}}{2\lambda}. \end{aligned} \tag{5.12}$$

For condition (iii) of Theorem 3.1 we need to compute $Var[\hat{Y}_{l,N_l}]$ and from above we have

$$\begin{aligned} Var[\hat{Y}_{l,N_l}] &= N_l^{-1} Var[\mathcal{P}_l - \mathcal{P}_{l-1}] \\ &= N_l^{-1} Var\left(X_{M_l}^{(f)} - X_{M_l}^{(c)}\right) \\ &\leq N_l^{-1} \tilde{C}\sigma^2 h_l^2 \frac{M_l}{2} \frac{1 - e^{-2\lambda h_l}}{2\lambda} \\ &= N_l^{-1} \tilde{C}\sigma^2 h_l^2 \frac{T h_l^{-1}}{2} \frac{1 - e^{-2\lambda h_l}}{2\lambda} \\ &= N_l^{-1} \frac{\tilde{C}\sigma^2 T}{2} \frac{1 - e^{-2\lambda h_l}}{2\lambda} h_l, \end{aligned}$$

where $M_l = T/h_l$ is the total number of timesteps on level l.

For small λh_l, expanding the exponential gives

$$Var[\hat{Y}_{l,N_l}] \leq N_l^{-1} \frac{\tilde{C}\sigma^2 T}{2} h_l^2 + \text{higher order terms}, \tag{5.13}$$

which proves condition (iii) and therefore completes the proof of Theorem 3.1. However if $\lambda h_l \gg 1$ then $1 - e^{-2\lambda h_l}$ is of order 1 so the variance is bounded by a $\mathcal{O}(h)$ term and does not necessarily decay with a quadratic rate.

□

In Figure 5-1 we demonstrate numerically the results from Lemma 5.1. We use the MLMC method with $L = 5$ number of levels. For the homogeneous model we use $\lambda = 500$, $\sigma = 1$ and $T = 1$ and we vary the size of M_0. Since at level l the fine timestep equals $h_l = \frac{T}{M_0 2^l}$ we would expect from the proof of Lemma 5.1 that as M_0 increases the variance rates will become quadratic. Indeed this is what we observe and when M_0 is small the variance rates are linear and as we increase M_0 they become quadratic.

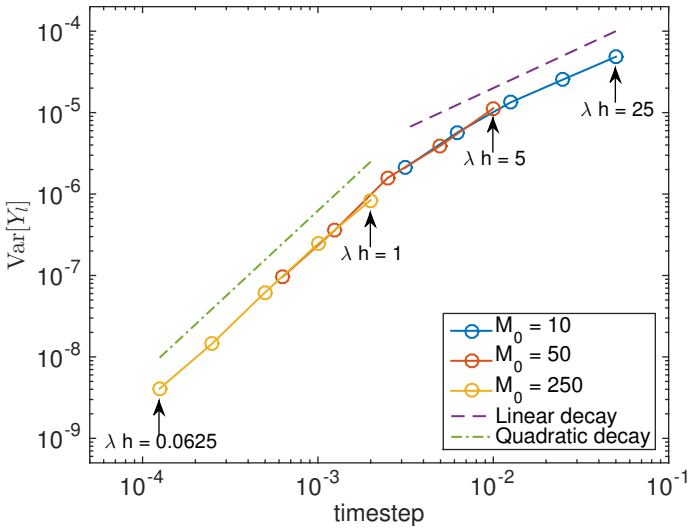


Figure 5-1: Variance rates with large λ (= 500)

5.2 Boundary conditions treatment

In Section 2.3 we introduced the models' boundary conditions and after defining our timestepping methods we can plot some particle trajectories and study the effect of reflection in the MLMC method. In Figure 5-2 we plot 50 trajectories using the Geometric Langevin method and the inhomogeneous model with $U(0) = 0.1$, $X(0) = 0.05$, $T = 1$, $M_0 = 10$ and $L = 5$. The boundary conditions are as in equations (2.26) and (2.27) and if the approximation X_n at time t_n becomes negative we replace both X_n and U_n with $-X_n$ and $-U_n$ respectively. Similarly if X_n becomes larger than H we replace both X_n and U_n with $2H - X_n$ and $-U_n$ respectively. From the plot we observe that the particles stay away from the top boundary because T is not very large.

The problem that now arises with reflection is that it is harder to keep the fine and coarse paths of a level of the MLMC method together since some particles are reflected and others do not. This leads to a reduction of the MLMC variance rates and a violation of condition (iii) in Theorem 3.1. In Figure 5-3 we plot the variance rates for the Symplectic Euler method which are not quadratic and are even worse than linear due to the incorrect coupling between fine and coarse paths. Therefore, before we study theoretically under which conditions Theorem 3.1 holds for the full model with reflection, it is necessary to construct an algorithm which gives a quadratic decay

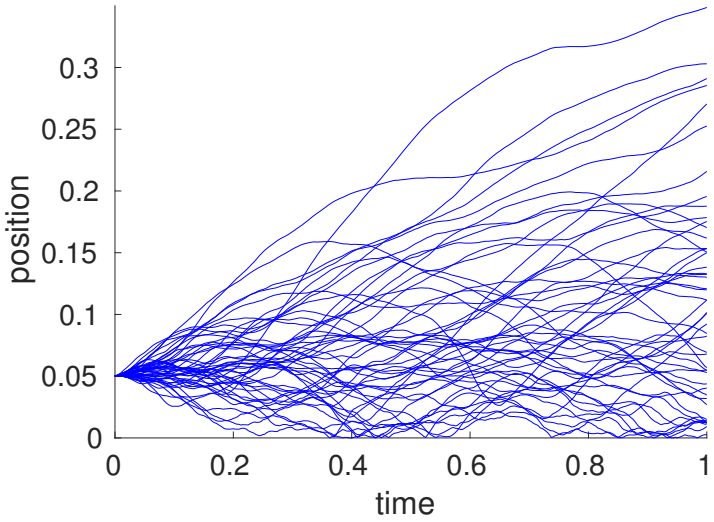


Figure 5-2: Trajectories plot

in the MLMC variance.

To recover quadratic rates and the correct coupling between fine and coarse paths we apply a new special treatment on the boundaries. This new methodology has also been published in Katsiolides et al. (2018). The key idea is to extend the domain of X to the whole real line and make it equivalent to a problem without boundary conditions. We firstly explain this idea for one reflective boundary and in Section 5.2.3 we show that the same treatment works for both boundaries by considering periodic extensions.

We begin by formulating an extended SDE in the variables $\tilde{X}$ and $\tilde{U}$ which is related to the initial one by

$$\begin{pmatrix} U \\ X \end{pmatrix} = S(\tilde{X}) \begin{pmatrix} \tilde{U} \\ \tilde{X} \end{pmatrix}, \tag{5.14}$$

where $S(\tilde{X(t)}) = (-1)^{\text{number of reflections up to time } t} = sign(\tilde{X}(t))$. In particular, for the general SDE problem

$$\begin{aligned} dU(t) &= a(X(t), U(t))dt + b(X(t))dW(t), \\ dX(t) &= U(t)dt, \end{aligned} \tag{5.15}$$

for some functions a and b with reflection only at $X = 0$, the extended SDE problem

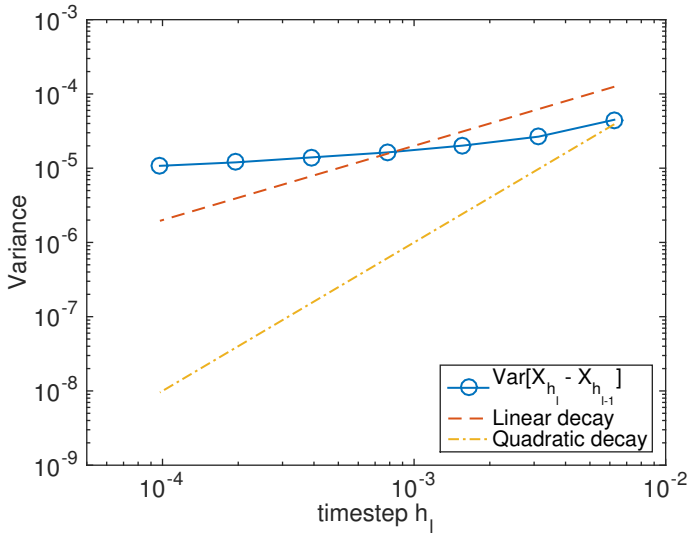


Figure 5-3: Symplectic Euler - Boundary Conditions and no special treatment
Mean Particle Position

is

$$\begin{aligned} d\tilde{U}(t) &= A(\tilde{X}(t), \tilde{U}(t))dt + B(\tilde{X}(t))dW(t), \\ d\tilde{X}(t) &= \tilde{U}(t)dt, \end{aligned} \tag{5.16}$$

where

$$A(\tilde{X}(t), \tilde{U}(t)) = \begin{cases} a(\tilde{X}(t), \tilde{U}(t)) & \text{if} \quad \tilde{X}(t) \geq 0, \\ -a(-\tilde{X}(t), -\tilde{U}(t)) & \text{if} \quad \tilde{X}(t) < 0, \end{cases} \tag{5.17}$$

and

$$B(\tilde{X}(t)) = \begin{cases} b(\tilde{X}(t)) & \text{if} \quad \tilde{X}(t) \geq 0, \\ b(-\tilde{X}(t)) & \text{if} \quad \tilde{X}(t) < 0. \end{cases} \tag{5.18}$$

The relation in (5.14) between $X(t)$ and $\tilde{X}(t)$ is also illustrated in Figure 5-4.

Having formulated the extended SDE makes it now possible to solve these new equations and then reconstruct the variables $U(t)$ and $X(t)$ using equation (5.14). However, after some analysis that we present below, it turns out that this is not necessary and you only need to multiply the random variables by a special coefficient in order to make the initial problem equivalent to the extended problem.

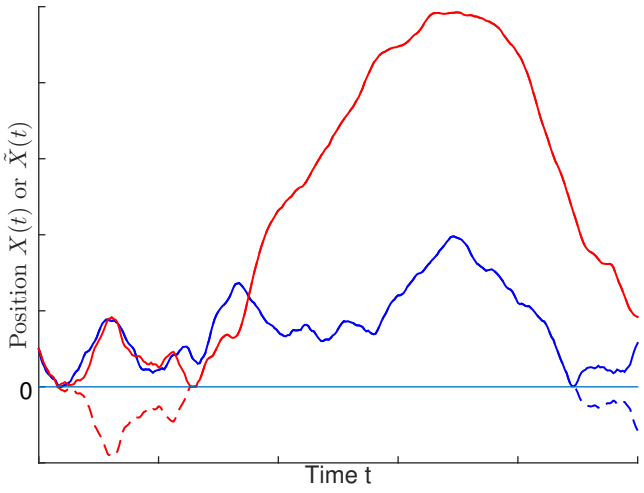


Figure 5-4: Trajectories of the original solution $X(t)$ (solid lines) and the extended solution $\tilde{X}(t)$ (dashed lines)

5.2.1 Boundary conditions treatment analysis

Here, we present the analysis for the Symplectic Euler method as in Katsiolides et al. (2018) and in Section 5.2.4 we present the analysis for the Geometric Langevin and BAOAB methods. Firstly, we derive some relations between the extended and the original variables and the SDEs in equations (5.15) and (5.16). Discretising equation (5.14) gives

$$\begin{pmatrix} U_n \\ X_n \end{pmatrix} = S_n \begin{pmatrix} \tilde{U}_n \\ \tilde{X}_n \end{pmatrix}, \tag{5.19}$$

where $S_n = (-1)^{\text{number of reflections up to time } t_n}$. Using this in equations (5.17) and (5.18), we obtain

$$A(\tilde{X}_n, \tilde{U}_n) = S_n a(S_n\tilde{X}_n, S_n\tilde{U}_n) = S_n a(X_n, U_n), \quad B(\tilde{X}_n) = b(S_n\tilde{X}_n) = b(X_n). \tag{5.20}$$

For a fine path on MLMC level with timestep size h, the Symplectic Euler method applied to the extended model in equation (5.16) gives

$$\begin{aligned} \tilde{U}_{n+1} &= \tilde{U}_n + A(\tilde{X}_n, \tilde{U}_n)\, h + B(\tilde{X}_n)\sqrt{h}\tilde{\xi}_n, \\ \tilde{X}_{n+1} &= \tilde{X}_n + \tilde{U}_{n+1} h, \end{aligned} \tag{5.21}$$

where $\tilde{\xi}_n \sim \text{Normal}(0,1)$ i.i.d. If we replace $(\tilde{U}_n, \tilde{X}_n)$ by (U_n, X_n) using equation (5.19) gives the following equations with the original variables U_n and X_n

$$\begin{aligned} S_{n+1}^{(f)} U_{n+1} &= S_n^{(f)} U_n + S_n^{(f)} a(X_n, U_n)\, h + b(X_n)\, \sqrt{h}\, \tilde{\xi}_n^{(f)}, \\ S_{n+1}^{(f)} X_{n+1} &= S_n^{(f)} X_n + S_{n+1}^{(f)}\, U_{n+1}\, h, \end{aligned} \tag{5.22}$$

The superscript (f) and later (c) is used in order to distinguish between fine and coarse MLMC steps. Multiplying both sides by $\left(S_{n+1}^{(f)}\right)^{-1} = S_{n+1}^{(f)}$ and defining $s_{n+1}^{(f)} = S_{n+1}^{(f)} S_n^{(f)}$, the timestepping scheme becomes

$$\begin{aligned} U_{n+1} &= s_{n+1}^{(f)} \left(U_n + a(X_n, U_n)\, h + b(X_n)\, \sqrt{h}\, S_n^{(f)}\, \tilde{\xi}_n^{(f)} \right), \\ X_{n+1} &= s_{n+1}^{(f)} \left(X_n + s_{n+1}^{(f)}\, U_{n+1}\, h \right). \end{aligned} \tag{5.23}$$

Note that $s_{n+1}^{(f)}$ equals -1 if there is a reflection between time t_n and t_{n+1} or 1 otherwise. Going back to the original SDEs in equation (5.15), discretising and implementing the reflection would give

$$\begin{aligned} U_{n+1} &= s_{n+1}^{(f)} \left(U_n + a(X_n, U_n)\, h + b(X_n)\, \sqrt{h}\, \xi_n^{(f)} \right), \\ X_{n+1} &= s_{n+1}^{(f)} \left(X_n + s_{n+1}^{(f)}\, U_{n+1}\, h \right), \end{aligned} \tag{5.24}$$

where $\xi_n^{(f)} \sim \text{Normal}(0,1)$ i.i.d. and we can now see that the original and extended discretisations are matched if

$$\xi_n^{(f)} = S_n^{(f)} \tilde{\xi}_n^{(f)}. \tag{5.25}$$

Since $\xi_n^{(f)}$ and $\tilde{\xi}_n^{(f)}$ agree in distribution, it means that in practice we can discretise the original SDEs and multiply each generated random variable $\xi_n^{(f)}$ by $S_n^{(f)}$. Similarly, in the next fine level we multiply the random variable $\xi_{n+1}^{(f)}$ by $S_{n+1}^{(f)}$. Exactly the same technique is applied at the coarse step where we use

$$\xi_n^{(c)} = S_n^{(c)} \tilde{\xi}_n^{(c)}. \tag{5.26}$$

To couple correctly the random variables in the MLMC coarse path of the extended

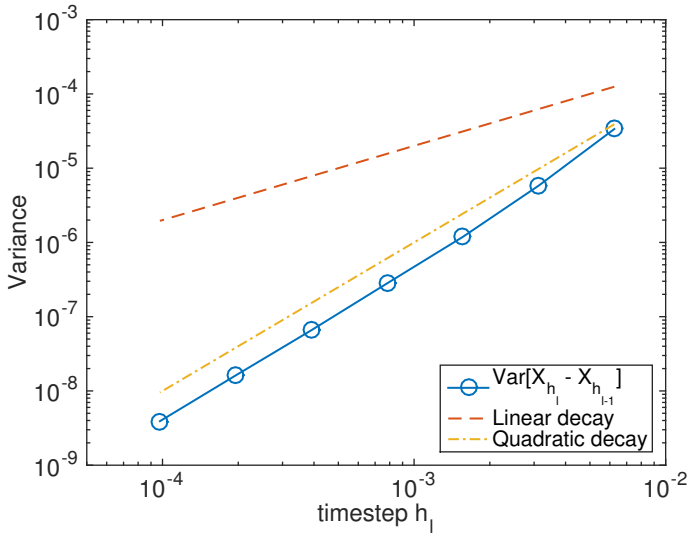


Figure 5-5: Symplectic Euler - Boundary Conditions with special treatment Mean Particle Position

model, the following expression is used (see also equation (3.25))

$$\tilde{\xi}_n^{(c)} = \frac{1}{\sqrt{2}}(\tilde{\xi}_n^{(f)} + \tilde{\xi}_{n+1}^{(f)}). \tag{5.27}$$

Then, by substituting equations (5.25) and (5.26) we can take into account the boundary conditions treatment by replacing equation (3.25) with

$$\xi_n^{(\text{coarse})} = \frac{S_n^{(c)}}{\sqrt{2}} \left(S_n^{(f)} \xi_n + S_{n+1}^{(f)} \xi_{n+1} \right). \tag{5.28}$$

Since multiplying by $S_n^{(\cdot)} = \pm 1$ does not change the distribution of the normal random variables, no extra bias is added and the telescoping sum (3.22) is preserved.

In Figure 5-5, we present the numerical result after we apply the above treatment and we observe that the quadratic variance rates are now recovered. For Geometric Langevin and BAOAB methods, a similar argument gives the same results as we show in Section 5.2.4.

5.2.2 Some comments on the discontinuities

Note that this treatment adds a discontinuity in the extended model since the function $A(X,U)$ is not continuous in general at $X=0$ (since $a(0,U)$ is not always equal to $-a(0,-U)$ in equation (5.17)). For the inhomogeneous model and the Met Office's constant regularisation (equations (2.24) and (2.25)) no discontinuities are introduced by this treatment since the $\frac{\partial V}{\partial X}(X,U)$ term disappears when $X=0$ and the $\lambda(X)U$ term stays continuous (in this case $a(X,U)=-\lambda(X)U-\frac{\partial V}{\partial X}(X,U)$). However, we have a discontinuity at $X=\epsilon_{\text{reg}}$ as shown in Figure 4-1a which is not a result of the boundary conditions treatment.

The opposite happens with the smooth regularisation described by the equations in (4.13) which is continuous at $X=\epsilon_{\text{reg}}$. The only discontinuity is at $X=0$ which is a result of the boundary conditions treatment. In Section 5.3.2 we describe an alternative type of smooth regularisation which gives continuous coefficients for both the original and the extended inhomogeneous models.

In Chapter 6 all our numerical results are done by using the Met Office's constant regularisation (equations (2.24) and (2.25)) and we didn't observe any problems due to the discontinuities. A theoretical study of the effect of these discontinuities can be a topic for future research. Finally, in Figure 5-6 we plot the extended inhomogeneous model coefficients with the Met Office's constant regularisation.

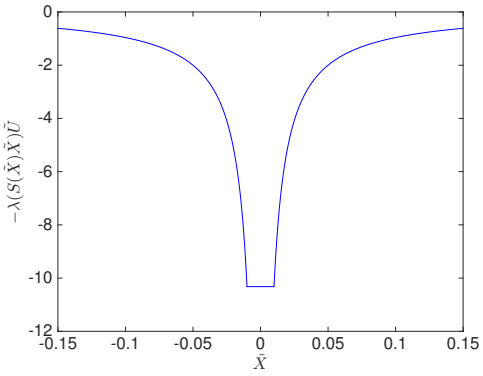
5.2.3 Reflection at the top boundary

For the top boundary reflection we consider periodic extensions of the domain. For $X\in\mathbb{R}$, let

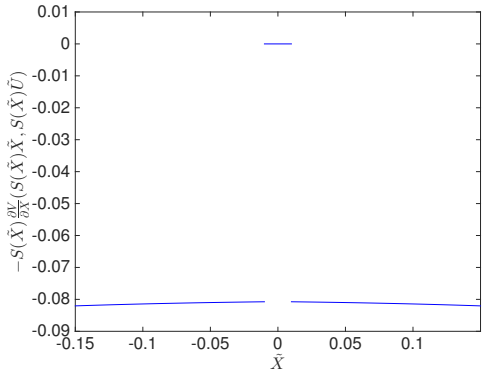
$$\begin{aligned}&\theta(X)=2n \text{ if } X\in[(2n-1)H,(2n+1)H), n\in\mathbb{Z},\\&\eta(X)=X-\theta(X)H.\end{aligned}$$

Then, similarly to equations (5.17) and (5.18) we define the extended coefficients as

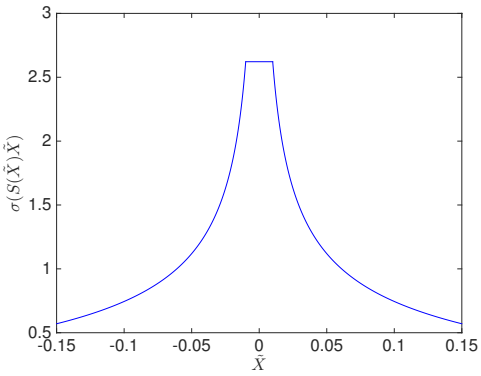
$$A(\tilde{X}(t),\tilde{U}(t))=\begin{cases}a(\eta(\tilde{X}(t)),\tilde{U}(t)) & \text{if} \quad \eta(\tilde{X}(t))\geq 0,\\ -a(-\eta(\tilde{X}(t)),-\tilde{U}(t)) & \text{if} \quad \eta(\tilde{X}(t))<0,\end{cases} \tag{5.29}$$



(a) $-\lambda(S(\tilde{X})\tilde{X})\tilde{U}$



(b) $-S(\tilde{X})\frac{\partial V}{\partial X}(S(\tilde{X})\tilde{X}, S(\tilde{X})\tilde{U})$



(c) $\sigma(S(\tilde{X})\tilde{X})$

Figure 5-6: Extended inhomogeneous model coefficients

and

$$B(\tilde{X}(t)) = \begin{cases} b(\eta(\tilde{X}(t))) & \text{if} \quad \eta(\tilde{X}(t)) \geq 0, \\ b(-\eta(\tilde{X}(t))) & \text{if} \quad \eta(\tilde{X}(t)) < 0. \end{cases} \tag{5.30}$$

It is now easy to see that $\eta(X)$ reduces the problem to the case of reflecting only at $X = 0$ since this function brings X to the interval $[-H, H]$. Therefore, the same analysis applies and no further changes are required in equations (5.28) and (5.41) where $S_n^{(\cdot)}$ now counts reflections at both boundaries.

5.2.4 Boundary conditions treatment for Geometric Langevin and BAOAB

Since Geometric Langevin and BAOAB are specialised methods for the Langevin equation we firstly consider an extended inhomogeneous turbulence model. This model will have the form of a Langevin equation. Again, we consider reflection only at the lower boundary and the top boundary treatment follows from periodic extensions as in Section 5.2.3. The results for the splitting methods form an extension of the analysis presented in Section 5.2.1.

The extended inhomogeneous turbulence model

Following the previous notation we can write the extended SDE coefficients in equations (5.17) and (5.18) as

$$A(\tilde{X}(t), \tilde{U}(t)) = S(\tilde{X}(t))a(S(\tilde{X}(t))\tilde{X}(t), S(\tilde{X}(t))\tilde{U}(t)), \tag{5.31}$$

$$B(\tilde{X}(t)) = b(S(\tilde{X}(t))\tilde{X}(t)). \tag{5.32}$$

For the inhomogeneous turbulence model we have

$$a(X, U) = -\lambda(X)U - \frac{\partial V}{\partial X}(X, U), \qquad b(X) = \sigma(X), \tag{5.33}$$

so the extended inhomogeneous turbulence model is

$$d\tilde{U}(t) = -\lambda(S(\tilde{X}(t))\tilde{X}(t))\tilde{U}(t)dt - S(\tilde{X}(t))\frac{\partial V}{\partial X}\left(S(\tilde{X}(t))\tilde{X}(t), S(\tilde{X}(t))\tilde{U}(t)\right)dt + \sigma\left(S(\tilde{X}(t))\tilde{X}(t)\right)dW(t), \tag{5.34}$$

$$d\tilde{X}(t) = \tilde{U}(t)dt. \tag{5.35}$$

Geometric Langevin

Firstly we consider the reflection treatment for the Geometric Langevin timestepping method. Since the extended inhomogeneous turbulence model equations (5.34) and (5.35) have the form of a Langevin equation, we can apply directly the Geometric Langevin discretisation scheme given by equations (3.8), (3.9) and (3.10). This gives

$$\begin{aligned}
\tilde{U}^{\star}_{n+1} &= e^{-\lambda(S_n\tilde{X}_n)h}\tilde{U}_n + \sigma(S_n\tilde{X}_n)a_h(S_n\tilde{X}_n)\tilde{\xi}_n,\\
\tilde{U}_{n+1} &= \tilde{U}^{\star}_{n+1} - S_n\frac{\partial V}{\partial X}\left(S_n\tilde{X}_n, S_n\tilde{U}^{\star}_{n+1}\right)h,\\
\tilde{X}_{n+1} &= \tilde{X}_n + \tilde{U}_{n+1}h,
\end{aligned}$$

where $a_h(S_n\tilde{X}_n) = \sqrt{\frac{1-e^{-2\lambda(S_n\tilde{X}_n)h}}{2\lambda(S_n\tilde{X}_n)}}$ and $\tilde{\xi}_n \sim Normal(0,1)$. This scheme can also be written as

$$\begin{aligned}
\tilde{U}_{n+1} =\ & e^{-\lambda(S_n\tilde{X}_n)h}\tilde{U}_n - S_n\frac{\partial V}{\partial X}\left(S_n\tilde{X}_n, S_n\left[e^{-\lambda(S_n\tilde{X}_n)h}\tilde{U}_n + \sigma(S_n\tilde{X}_n)a_h(S_n\tilde{X}_n)\tilde{\xi}_n\right]\right)h\\
&+ \sigma(S_n\tilde{X}_n)a_h(S_n\tilde{X}_n)\tilde{\xi}_n,\\
\tilde{X}_{n+1} =\ & \tilde{X}_n + \tilde{U}_{n+1}h.
\end{aligned}$$

Using equation (5.19),

$$\begin{aligned}
S_{n+1}U_{n+1} =\ & e^{-\lambda(X_n)h}S_nU_n - S_n\frac{\partial V}{\partial X}\left(X_n, S_n\left[e^{-\lambda(X_n)h}S_nU_n + \sigma(X_n)a_h(X_n)\tilde{\xi}_n\right]\right)h\\
&+ \sigma(X_n)a_h(X_n)\tilde{\xi}_n,\\
S_{n+1}X_{n+1} =\ & S_nX_n + S_{n+1}U_{n+1}h.
\end{aligned}$$

Letting $s_{n+1} = S_{n+1}S_n$ we obtain the Geometric Langevin discretisation scheme for

the extended model

$$U_{n+1} = s_{n+1}\left[e^{-\lambda(X_n)h}U_n - \frac{\partial V}{\partial X}\left(X_n, e^{-\lambda(X_n)h}U_n + \sigma(X_n)a_h(X_n)S_n\tilde{\xi}_n\right)h + \sigma(X_n)a_h(X_n)S_n\tilde{\xi}_n\right], \tag{5.36}$$

$$X_{n+1} = s_{n+1}\left(X_n + s_{n+1}U_{n+1}h\right). \tag{5.37}$$

Note that s_{n+1} equals -1 if there is a reflection between time t_n and t_{n+1} or 1 otherwise.

The Geometric Langevin discretisation scheme for the original model with reflection is given by

$$U_{n+1} = s_{n+1}\left[e^{-\lambda(X_n)h}U_n - \frac{\partial V}{\partial X}\left(X_n, e^{-\lambda(X_n)h}U_n + \sigma(X_n)a_h(X_n)\xi_n\right)h + \sigma(X_n)a_h(X_n)\xi_n\right], \tag{5.38}$$

$$X_{n+1} = s_{n+1}\left(X_n + s_{n+1}U_{n+1}h\right), \tag{5.39}$$

where $\xi_n \sim Normal(0,1)$.

Comparing equations (5.36), (5.37) with equations (5.38), (5.39) we can see that the original and extended discretisations are matched if

$$\xi_n = S_n\tilde{\xi}_n, \tag{5.40}$$

which is the same as in the case of Symplectic Euler method. Then, by following the same analysis as in Section (5.2.1) and by considering the fine and coarse MLMC steps (and the same arguments after equation (5.25)) we deduce that the same treatment works for the Geometric Langevin method. Therefore, we only need to replace equation (3.26) with

$$\xi_n^{(\text{coarse})} = S_n^{(c)}\frac{e^{-\lambda(X_n)h}S_n^{(f)}\xi_n + S_{n+1}^{(f)}\xi_{n+1}}{\sqrt{e^{-2\lambda(X_n)h}+1}}, \tag{5.41}$$

where $S_n^{(f)}$ and $S_n^{(c)}$ count reflections on the fine and coarse steps respectively.

BAOAB

As in the Geometric Langevin case, we can discretise directly the extended inhomogeneous model using the BAOAB method given by equations (3.12) - (3.16). This gives

$$
\begin{aligned}
\tilde{U}_{n+\frac{1}{2}} &= \tilde{U}_n - S_n \frac{\partial V}{\partial X}\left(S_n\tilde{X}_n, S_n\tilde{U}_n\right)\frac{h}{2},\\
\tilde{X}_{n+\frac{1}{2}} &= \tilde{X}_n + \tilde{U}_{n+\frac{1}{2}}\frac{h}{2},\\
\tilde{U}^{\star}_{n+\frac{1}{2}} &= e^{-\lambda(S_{n+\frac{1}{2}}\tilde{X}_{n+\frac{1}{2}})h}\tilde{U}_{n+\frac{1}{2}} + \sigma(S_{n+\frac{1}{2}}\tilde{X}_{n+\frac{1}{2}})a_h(S_{n+\frac{1}{2}}\tilde{X}_{n+\frac{1}{2}})\tilde{\xi}_n,\\
\tilde{X}_{n+1} &= \tilde{X}_{n+\frac{1}{2}} + \tilde{U}^{\star}_{n+\frac{1}{2}}\frac{h}{2},\\
\tilde{U}_{n+1} &= \tilde{U}^{\star}_{n+\frac{1}{2}} - S_{n+1}\frac{\partial V}{\partial X}\left(S_{n+1}\tilde{X}_{n+1}, S_{n+1}\tilde{U}^{\star}_{n+\frac{1}{2}}\right)\frac{h}{2},
\end{aligned}
$$

where $\tilde{\xi}_n \sim Normal(0,1)$. Using equation (5.19) it becomes

$$
\begin{aligned}
S_{n+\frac{1}{2}}U_{n+\frac{1}{2}} &= S_nU_n - S_n\frac{\partial V}{\partial X}\left(X_n, U_n\right)\frac{h}{2},\\
S_{n+\frac{1}{2}}X_{n+\frac{1}{2}} &= S_nX_n + S_{n+\frac{1}{2}}U_{n+\frac{1}{2}}\frac{h}{2},\\
S_{n+1}U^{\star}_{n+\frac{1}{2}} &= e^{-\lambda(X_{n+\frac{1}{2}})h}S_{n+\frac{1}{2}}U_{n+\frac{1}{2}} + \sigma(X_{n+\frac{1}{2}})a_h(X_{n+\frac{1}{2}})\tilde{\xi}_n,\\
S_{n+1}X_{n+1} &= S_{n+\frac{1}{2}}X_{n+\frac{1}{2}} + S_{n+1}U^{\star}_{n+\frac{1}{2}}\frac{h}{2},\\
S_{n+1}U_{n+1} &= S_{n+1}U^{\star}_{n+\frac{1}{2}} - S_{n+1}\frac{\partial V}{\partial X}\left(X_{n+1}, U^{\star}_{n+\frac{1}{2}}\right)\frac{h}{2}.
\end{aligned}
$$

These equations now imply

$$
\begin{aligned}
U_{n+\frac{1}{2}} &= S_{n+\frac{1}{2}}S_n\left[U_n - \frac{\partial V}{\partial X}(X_n, U_n)\frac{h}{2}\right],\\
X_{n+\frac{1}{2}} &= S_{n+\frac{1}{2}}S_n\left[X_n + S_{n+\frac{1}{2}}S_nU_{n+\frac{1}{2}}\frac{h}{2}\right],\\
U^{\star}_{n+\frac{1}{2}} &= S_{n+1}S_{n+\frac{1}{2}}\left[e^{-\lambda(X_{n+\frac{1}{2}})h}U_{n+\frac{1}{2}} + \sigma(X_{n+\frac{1}{2}})a_h(X_{n+\frac{1}{2}})S_{n+\frac{1}{2}}\tilde{\xi}_n\right],\\
X_{n+1} &= S_{n+1}S_{n+\frac{1}{2}}\left[X_{n+\frac{1}{2}} + S_{n+1}S_{n+\frac{1}{2}}U^{\star}_{n+\frac{1}{2}}\frac{h}{2}\right],\\
U_{n+1} &= U^{\star}_{n+\frac{1}{2}} - \frac{\partial V}{\partial X}\left(X_{n+1}, U^{\star}_{n+\frac{1}{2}}\right)\frac{h}{2}.
\end{aligned}
$$

The discretisation scheme for the original model with reflection is

$$
\begin{aligned}
U_{n+\frac{1}{2}} &= S_{n+\frac{1}{2}} S_n \left[U_n - \frac{\partial V}{\partial X}(X_n, U_n) \frac{h}{2} \right], \\
X_{n+\frac{1}{2}} &= S_{n+\frac{1}{2}} S_n \left(X_n + S_{n+\frac{1}{2}} S_n U_{n+\frac{1}{2}} \frac{h}{2} \right), \\
U^{\star}_{n+\frac{1}{2}} &= S_{n+1} S_{n+\frac{1}{2}} \left[e^{-\lambda(X_{n+\frac{1}{2}})h} U_{n+\frac{1}{2}} + \sigma(X_{n+\frac{1}{2}}) a_h(X_{n+\frac{1}{2}}) \xi_n \right], \\
X_{n+1} &= S_{n+1} S_{n+\frac{1}{2}} \left(X_{n+\frac{1}{2}} + S_{n+1} S_{n+\frac{1}{2}} U^{\star}_{n+\frac{1}{2}} \frac{h}{2} \right), \\
U_{n+1} &= U^{\star}_{n+\frac{1}{2}} - \frac{\partial V}{\partial X} \left(X_{n+1}, U^{\star}_{n+\frac{1}{2}} \right) \frac{h}{2},
\end{aligned}
$$

where $\xi_n \sim Normal(0, 1)$.

Comparing the discretisation schemes for the original and extended models we can see that they are matched if

$$\xi_n = S_{n+\frac{1}{2}} \tilde{\xi}_n. \tag{5.42}$$

The variable $S_{n+\frac{1}{2}}$ counts the number of reflections just before the Ornstein-Uhlenbeck step. Equation (5.42) was also derived in the Geometric Langevin case and implies that the same treatment works. Therefore, we use the random variable

$$\xi_n^{(\text{coarse})} = S_n^{(c)} \frac{e^{-\lambda(X_n)h} S_n^{(f)} \xi_n + S_{n+1}^{(f)} \xi_{n+1}}{\sqrt{e^{-2\lambda(X_n)h} + 1}}, \tag{5.43}$$

for the MLMC coarse step.

5.2.5 Some final comments

As a final note in the boundary conditions treatment section, we briefly discuss a problem that might appear when discretising SDEs with reflection, as described in Ramli and Esler (2016). The authors of Ramli and Esler (2016) studied a problem arising with reflective boundary conditions when timestepping methods with intermediate steps (like for example Geometric Langevin and BAOAB) are used to discretise the inhomogeneous model equations (2.7) and (2.8). In particular, when the timestep size was not small enough, some errors due to reflective boundary conditions arised that dominated the discretisation and sampling errors of their higher order schemes that involve intermediate timesteps.

Since this problem was about timestepping methods with intermediate steps, it is possible that it might also affect our timestepping methods and in particular Geometric Langevin and BAOAB. In order to solve this issue, a simple method is described in Ramli and Esler (2016). Firstly, at the start of each timestep, the domain of X is extended to $(-\infty, \infty)$ by repeated reflection of $\sigma_U(x)$ and $\tau(x)$. All intermediate steps are then completed in the extended domain and at the end of the full timestep the particle is taken back in $[0, 1]$ using reflection as described in the first paragraph of Section 5.2. This treatment then removes any problems with reflective boundary conditions.

In our numerical experiments we did not observe this issue when using any of the two timestepping methods and therefore it was not necessary to apply the method described in Ramli and Esler (2016). Further investigation however on this problem and how it affects MLMC performance can be a topic for future work.

5.3 Inhomogeneous model and the complexity theorem

We now study the theoretical application of Theorem 3.1 (complexity theorem, Giles (2008)) to the inhomogeneous model with reflection and in particular to the extended model as described in Section 5.2. A direct approach to apply Theorem 3.1 as we did for the homogeneous model in Section 5.1 is very difficult and challenging due to the fact that the inhomogeneous model is a non-linear SDE. Here, we study an alternative approach based on the theory of modified equations (Shardlow, 2006), (Zygalakis, 2011) and their application to the MLMC method. As described in Müller et al. (2015), modified equations provide an alternative method to check whether the conditions of Theorem 3.1 hold which does not directly rely on strong approximation results.

5.3.1 Modified equations and the complexity theorem

We begin by defining the modified equations (Shardlow, 2006), (Zygalakis, 2011). Consider the general SDE

$$d\mathbf{X}(t) = \mathbf{f}(\mathbf{X}(t))dt + G(\mathbf{X}(t))d\mathbf{W}(t), \tag{5.44}$$

with $\mathbf{X}(t) \in \mathbb{R}^d, \mathbf{f} : \mathbb{R}^d \to \mathbb{R}^d, G : \mathbb{R}^d \to \mathbb{R}^{d\times m}$ and initial conditions $\mathbf{X}(0)$. Suppose that for some timestepping method (eg Euler method) we can compute a first order weak approximation $\mathbf{X}_M (M = T/h)$ of $\mathbf{X}(T)$, i.e.

$$|\mathbb{E}\left[\phi(\mathbf{X}(T))\right] - \mathbb{E}\left[\phi(\mathbf{X}_M)\right]| \leq Ch, \tag{5.45}$$

for some constant $C > 0$ and for all suitable test functions $\phi : \mathbb{R}^d \to \mathbb{R}$ (this can be generalised to p^{th} order). Under some conditions (Shardlow, 2006), (Zygalakis, 2011) it is then possible to compute functions $\mathbf{f}_1 : \mathbb{R}^d \to \mathbb{R}^d$ and $G_1 : \mathbb{R}^d \to \mathbb{R}^{d\times m}$ and construct a modified SDE

$$d\mathbf{X}^h(t) = \left(\mathbf{f}(\mathbf{X}^h(t)) + h\mathbf{f}_1(\mathbf{X}^h(t))\right) dt + \left(G(\mathbf{X}^h(t)) + hG_1(\mathbf{X}^h(t))\right) d\mathbf{W}(t), \tag{5.46}$$

such that the approximation $\mathbf{X}_M$ is a second order weak approximation to the exact solution $\mathbf{X}^h(t)$ of (5.46), i.e.

$$\left|\mathbb{E}\left[\phi(\mathbf{X}^h(T))\right] - \mathbb{E}\left[\phi(\mathbf{X}_M)\right]\right| \leq \tilde{C}h^2, \tag{5.47}$$

for some constant $\tilde{C} > 0$ and test functions ϕ. The functions $\mathbf{f}_1$ and G_1 depend on the coefficient functions in equation (5.44) and on the timestepping method.

The most important property of the modified equations is that with a small perturbation of the original SDE we can construct a new system of equations such that the timestepping method approximates its solution more accurately. This property is one of the key ideas considered in Müller et al. (2015) in order to provide an alternative approach for the application of Theorem 3.1.

In particular, in Müller et al. (2015) they consider the following doubled-up system of SDEs

$$d\mathbf{X}(t) = \mathbf{f}(\mathbf{X}(t))dt + G(\mathbf{X}(t))d\mathbf{W}(t), \tag{5.48}$$

$$d\mathbf{Y}(t) = \mathbf{f}(\mathbf{Y}(t))dt + G(\mathbf{Y}(t))d\mathbf{W}(t), \quad \mathbf{Y}(0) = \mathbf{X}(0). \tag{5.49}$$

The two equations have the same initial conditions and Brownian motion $\mathbf{W}(t)$. Equation (5.48) will be discretised with a timestep size $h/2$ as in the MLMC fine step and equation (5.49) will be discretised with a timestep size h as in the MLMC coarse step. The reason why we discretise with both the fine and coarse schemes is to add informa-

tion about the random variable coupling (eg equation (3.26) for Geometric Langevin) in the MLMC coarse step. As we already discussed in Section 3.2.2, the random variable coupling affects property (iii) of Theorem 3.1 (quadratic variance decay rate) and therefore it is necessary to be taken into account in the proof. Then, following the technique from Zygalakis (2011) we compute modified equations of the form

$$d\mathbf{X}^h(t) = \big(\mathbf{f}(\mathbf{X}^h(t)) + h\mathbf{f}_1(\mathbf{X}^h(t))\big)\,dt + \big(G(\mathbf{X}^h(t)) + hG_1(\mathbf{X}^h(t))\big)\,d\mathbf{W}(t), \tag{5.50}$$

$$d\mathbf{Y}^h(t) = \big(\mathbf{f}(\mathbf{Y}^h(t)) + h\mathbf{f}_2(\mathbf{Y}^h(t))\big)\,dt + \big(G(\mathbf{Y}^h(t)) + hG_2(\mathbf{Y}^h(t))\big)\,d\mathbf{W}(t), \tag{5.51}$$

with initial conditions $\mathbf{X}^h(0) = \mathbf{X}(0), \mathbf{Y}^h(0) = \mathbf{X}(0)$. The functions $\mathbf{f}_i : \mathbb{R}^d \to \mathbb{R}^d$ and $G_i : \mathbb{R}^d \to \mathbb{R}^{d\times m}$ for $i = 1, 2$. Also, let $\mathbf{Z}(t) = [\mathbf{X}(t), \mathbf{Y}(t)]$ be the exact solution of equations (5.48) and (5.49) and denote its approximation by $\mathbf{Z}_n = [\mathbf{X}_n, \mathbf{Y}_n]$. The following lemma is then proved in Müller et al. (2015) (Lemma 3.3).

Lemma 5.2. *For $t \in [0, T]$, let $\mathbf{Z}(t)$ satisfy the SDEs* (5.48) *and* (5.49) *and let $\mathbf{Z}^h(t) = [\mathbf{X}^h(t), \mathbf{Y}^h(t)]$ satisfy the modified equations* (5.50) *and* (5.51)*. Suppose that*

(i) $\mathbf{f} : \mathbb{R}^d \to \mathbb{R}^d$ and $G : \mathbb{R}^d \to \mathbb{R}^{d\times m}$ are globally Lipschitz continuous with Lipschitz constant $L > 0$.

(ii) There exists $C_1 > 0$ such that, for all $h > 0$ suffciently small,

$$\mathbb{E}\left[\left\|\mathbf{f}_1(\mathbf{X}^h(s))\right\|^2\right], \mathbb{E}\left[\left\|G_1(\mathbf{X}^h(s))\right\|_F^2\right] \le C_1,$$
$$\mathbb{E}\left[\left\|\mathbf{f}_2(\mathbf{Y}^h(s))\right\|^2\right], \mathbb{E}\left[\left\|G_2(\mathbf{Y}^h(s))\right\|_F^2\right] \le C_1, \quad s \in [0, T],$$

where $\|\cdot\|_F$ denotes the Frobenius norm.

Then, if $\psi : \mathbb{R}^{2d} \to \mathbb{R}$ is globally Lipschitz continuous, we have for some constant $C_2 > 0$ independent of h,

$$\mathbb{E}\left[\left|\psi(\mathbf{Z}(t)) - \psi(\mathbf{Z}^h(t))\right|^2\right] \le C_2h^2, for\ t \in [0, T]. \tag{5.52}$$

As it is noted in Müller et al. (2015), a sufficient condition for the regularity assumptions of Lemma 5.2 to hold, is that the coefficients $\mathbf{f}, \mathbf{f}_i, G$ and G_i are globally

Lipschitz continuous.

In the next theorem, we can see how the modified equations and the regularity conditions of Lemma 5.2 can be used to show that Theorem 3.1 holds.

Theorem 5.3. *Fix $T > 0$. Let $\phi : \mathbb{R}^d \to \mathbb{R}$ be globally Lipschitz continuous and infinitely differentiable with polynomially bounded derivatives. Suppose that*

(i) $\mathbf{X}_n$ and $\mathbf{Y}_n$ are weak order-α approximations to $\mathbf{X}(t)$ for some $\alpha > 1/2$ (see also Definition 4.11),

(ii) $\mathbf{Z}_n$ are second-order weak approximations to $\mathbf{Z}^h(t)$ and

(iii) the assumptions of Lemma 5.2 hold.

Then Conditions $(i) - (iii)$ of Theorem 3.1 (MLMC complexity theorem) hold with $[\mathcal{P}_l^{(i)}, \mathcal{P}_{l-1}^{(i)}]$ given by i.i.d. samples of $[\phi(\mathbf{X}_{M_l}), \phi(\mathbf{Y}_{M_l})]$ with $h = h_l$ and $(\mathbf{X}_{M_l}, \mathbf{Y}_{M_l})$ the approximation of $(\mathbf{X}_T, \mathbf{Y}_T)$.

$Proof$. See Theorem 3.4 in Müller et al. (2015).

The conclusion of Theorem 5.3 is that with the existence of a second-order modified equation and with some regularity conditions on the SDE coefficients we can prove that the conditions of Theorem 3.1 hold. Therefore, as a result of Theorem 3.1, we expect to see a $\mathcal{O}(\epsilon^{-2})$ cost rate for MLMC that improves the corresponding cost rate of StMC which is $\mathcal{O}(\epsilon^{-3})$ (see Section 3.2.1 for the StMC cost rate). In the next section, we apply this modified equations theory to the extended inhomogeneous model equations (5.34) and (5.35) and study under which conditions Theorem 3.1 holds.

5.3.2 Application to the inhomogeneous model

We are now going to study the application of the modified equations theory to the extended inhomogeneous model described in Section 5.2. We only consider reflection at the lower boundary but a similar approach also works for the top boundary. For convenience we begin by writing down the extended inhomogeneous model as derived in Section 5.2.4. This is given by

$$\begin{aligned} d\tilde{U}(t) = &- \lambda(S(\tilde{X}(t))\tilde{X}(t))\tilde{U}(t)dt - S(\tilde{X}(t))\frac{\partial V}{\partial X}\left(S(\tilde{X}(t))\tilde{X}(t), S(\tilde{X}(t))\tilde{U}(t)\right)dt \\ &+ \sigma\left(S(\tilde{X}(t))\tilde{X}(t)\right)dW(t), \end{aligned} \tag{5.53}$$

$$d\tilde{X}(t) = \tilde{U}(t)dt. \tag{5.54}$$

where $S(\tilde{X}(t)) = (-1)^{\text{number of reflections up to time } t} = sign(\tilde{X}(t))$.

As Theorem 5.3 suggests, we firstly look for second order modified equations of (5.53) and (5.54) with $\lambda(X)$, $\frac{\partial V}{\partial X}(X,U)$ and $\sigma(X)$ defined by (2.9) and (2.10). In Zygalakis (2011) a general technique for deriving modified equations is described for SDEs with smooth coefficients. In particular, for a first order timestepping method it is possible in some cases to derive a second order modified equation (can be generalised to higher orders).

Looking at equations (5.53) and (5.54) we can easily see that the SDE coefficients are not smooth and some of them are also discontinuous and the reason for this is the presence of the function $S(\tilde{X}(t)) = sign(\tilde{X}(t))$. To solve this problem we can apply a special regularisation of the functions $\sigma_U(X)$ and $\tau(X)$ in the SDE coefficients

$$\lambda(X) = \frac{1}{\tau(X)}, \quad \sigma(X) = \sqrt{\frac{2\sigma_U^2(X)}{\tau(X)}}, \tag{5.55}$$

$$\frac{\partial V(X,U)}{\partial X} = -\frac{1}{2}\left[1 + \left(\frac{U}{\sigma_U(X)}\right)^2\right]\frac{\partial \sigma_U^2(X)}{\partial X}. \tag{5.56}$$

As described in Section 2.2, regularising the SDE coefficients means to replace them with some similar functions in order to make sure that the timestepping methods remain stable. In addition, it can make the model easier to analyse theoretically. Instead of using a constant regularisation as the Met Office does (equations (2.24) and (2.25)), we replace $\sigma_U(X)$ and $\tau(X)$ with some functions $\sigma_U^{\epsilon_{\text{reg}}}(X)$ and $\tau^{\epsilon_{\text{reg}}}(X)$ on the intervals $[0, \epsilon_{\text{reg}}]$ and $[H - \epsilon_{\text{reg}}, H]$. These functions have the following properties

$$\tau(\epsilon_{\text{reg}}) = \tau^{\epsilon_{\text{reg}}}(\epsilon_{\text{reg}}), \quad \frac{\partial \tau}{\partial X}(\epsilon_{\text{reg}}) = \frac{\partial \tau^{\epsilon_{\text{reg}}}}{\partial X}(\epsilon_{\text{reg}}), \quad \frac{\partial \tau^{\epsilon_{\text{reg}}}}{\partial X}(0) = 0, \tag{5.57}$$

and

$$\sigma_U(\epsilon_{\text{reg}}) = \sigma_U^{\epsilon_{\text{reg}}}(\epsilon_{\text{reg}}), \quad \frac{\partial \sigma_U}{\partial X}(\epsilon_{\text{reg}}) = \frac{\partial \sigma_U^{\epsilon_{\text{reg}}}}{\partial X}(\epsilon_{\text{reg}}), \tag{5.58}$$

$$\frac{\partial \sigma_U^{\epsilon_{\text{reg}}}}{\partial X}(0) = 0, \quad \frac{\partial^2 \sigma_U}{\partial X^2}(\epsilon_{\text{reg}}) = \frac{\partial^2 \sigma_U^{\epsilon_{\text{reg}}}}{\partial X^2}(\epsilon_{\text{reg}}), \tag{5.59}$$

with similar changes for the upper interval. These conditions guarantee that the coefficients of the extended model are continuously differentiable. If we need continuity for higher order derivatives then we can just add similar conditions on higher order derivatives of $\sigma_U^{\epsilon_{\text{reg}}}(X)$ and $\tau^{\epsilon_{\text{reg}}}(X)$. For simplicity we can assume that $\sigma_U^{\epsilon_{\text{reg}}}(X)$ and $\tau^{\epsilon_{\text{reg}}}(X)$

are polynomials of sufficiently high degree and use the above conditions to solve a simultaneous system of linear equations that will give the polynomial coefficients.

With this choice of regularisation we satisfy the first condition from Zygalakis (2011) that requires SDEs with smooth coefficients. The second condition is the existence of a timestepping method which is weakly convergent with order 1. From Kloeden and Platen (2011) we know that the Euler-Maruyama method and consequently all the other methods considered in this thesis as discussed in Section 4.3.2, converge weakly with order 1 if the SDE coefficients are Lipschitz continuous, satisfy linear growth and have polynomially bounded derivatives up to order 4. So far we only have smoothness of the coefficients, but all other requirements will follow if we also apply the special transformation from Milstein and Tretyakov (2005) given by equation (4.47) which will give us coefficients with bounded derivarives. The combination of our special regularisation (according to equations (5.57), (5.58) and (5.59)) with the transformation from Milstein and Tretyakov (2005) results in SDE coefficients which are continuously differentiable and bounded up to a sufficiently high degree. As a result, the SDE coefficients will become Lipschitz continuous (this follows from the Mean Value Theorem) which means that our timestepping methods converge weakly with order 1. According to Proposition 4.12, approximating this new regularised SDE only adds a small error.

For the rest of this chapter we assume that the coefficients of the extended model equations (5.53) and (5.54) are Lipschitz continuous and have polynomially bounded derivatives up to order 4. To simplify the notation we rewrite the extended model as

$$d\bar{U}(t) = -\bar{\lambda}(\bar{X}(t))\bar{U}(t)dt - \frac{\partial \bar{V}}{\partial X}(\bar{X}(t), \bar{U}(t))dt + \bar{\sigma}(\bar{X}(t))dW(t), \tag{5.60}$$

$$d\bar{X}(t) = \bar{U}(t)dt, \tag{5.61}$$

where $\bar{\lambda}, \bar{\sigma}$ and $\frac{\partial \bar{V}}{\partial X}$ take into account all the necessary regularisations and transformations on the coefficients as discussed above. $(\bar{X}(t), \bar{U}(t))$ will be the solution of the extended model after we apply these changes on the coefficients. Finally, we can apply the analysis from Zygalakis (2011) to compute modified equations of the doubled-up system

$$d\bar{U}(t) = -\bar{\lambda}(\bar{X}(t))\bar{U}(t)dt - \frac{\partial \bar{V}}{\partial X}(\bar{X}(t), \bar{U}(t))dt + \bar{\sigma}(\bar{X}(t))dW(t), \tag{5.62}$$

$$d\bar{X}(t) = \bar{U}(t)dt, \tag{5.63}$$

$$d\bar{u}(t) = -\bar{\lambda}(\bar{x}(t))\bar{u}(t)dt - \frac{\partial \bar{V}}{\partial x}(\bar{x}(t), \bar{u}(t))dt + \bar{\sigma}(\bar{x}(t))dW(t), \tag{5.64}$$

$$d\bar{x}(t) = \bar{u}(t)dt. \tag{5.65}$$

The modified equations have the form

$$\begin{aligned} d\bar{U}(t) = &\left(-\bar{\lambda}(\bar{X}(t))\bar{U}(t) - \frac{\partial \bar{V}}{\partial X}(\bar{X}(t), \bar{U}(t)) + hv_1^{(\bar{U})}(\bar{X}(t), \bar{U}(t))\right) dt \\ &+ \left(\bar{\sigma}(\bar{X}(t)) + h\Sigma_1^{(\bar{U})}(\bar{X}(t), \bar{U}(t))\right) dW(t), \end{aligned} \tag{5.66}$$

$$d\bar{X}(t) = \left(\bar{U}(t) + hv_1^{(\bar{X})}(\bar{X}(t), \bar{U}(t))\right) dt + h\Sigma_1^{(\bar{X})}(\bar{X}(t), \bar{U}(t))dW(t), \tag{5.67}$$

$$\begin{aligned} d\bar{u}(t) = &\left(-\bar{\lambda}(\bar{x}(t))\bar{u}(t) - \frac{\partial \bar{V}}{\partial x}(\bar{x}(t), \bar{u}(t)) + hv_2^{(\bar{u})}(\bar{x}(t), \bar{u}(t))\right) dt \\ &+ \left(\bar{\sigma}(\bar{x}(t)) + h\Sigma_2^{(\bar{u})}(\bar{x}(t), \bar{u}(t))\right) dW(t), \end{aligned} \tag{5.68}$$

$$d\bar{x}(t) = \left(\bar{u}(t) + hv_2^{(\bar{x})}(\bar{x}(t), \bar{u}(t))\right) dt + h\Sigma_2^{(\bar{x})}(\bar{x}(t), \bar{u}(t))dW(t). \tag{5.69}$$

Due to the assumptions we have just made, the analysis applies for our three timestepping methods and the modified equations can be computed using a computer algebra system. Here we give a brief description of the main steps and the reader is referred to Zygalakis (2011) for the full details. To compute the coefficients of the modified equations we developed Python code that can be downloaded from `https://bitbucket.org/em459/modified-equations`. When the coefficients are computed they are extracted to a LaTeX file which can also be found in the above link. The names of some of the basic classes from this code are used in the text below in order to explain how the modified equations are computed. Note that the code can also compute modified equations for a single level timestepping method but in this thesis we focus only on doubled-up systems of equations. The user can compute a single system or a doubled-up system of modified equations by either choosing the class TimestepSinglelevel() or the class TimestepMultilevel().

Following the technique from Zygalakis (2011), the first step is to compute the generator $\mathcal{L}_0$ of the doubled-up system of equations (5.62)-(5.65) (this is done with the function L0($\cdot$) from the Python code). For our SDEs the generator $\mathcal{L}_0$ is given by the following definition

Definition 5.4. *The generator $\mathcal{L}_0$ of the doubled-up system of equations* (5.62)-(5.65) *is defined by*

$$\begin{aligned}\mathcal{L}_0 F = &-\left(\bar{\lambda}(X)U + \frac{\partial \bar{V}}{\partial X}(X,U)\right)\frac{\partial F}{\partial U} + U\frac{\partial F}{\partial X} + \frac{\bar{\sigma}^2(X)}{2}\frac{\partial^2 F}{\partial U^2}\\ &-\left(\bar{\lambda}(x)u + \frac{\partial \bar{V}}{\partial x}(x,u)\right)\frac{\partial F}{\partial u} + u\frac{\partial F}{\partial x} + \frac{\bar{\sigma}^2(x)}{2}\frac{\partial^2 F}{\partial u^2}\\ &+\bar{\sigma}(X)\bar{\sigma}(x)\frac{\partial^2 F}{\partial U \partial u},\end{aligned} \tag{5.70}$$

where $F(X,U,x,u)$ is a twice differentiable function.

The second step is to compute the operator $\mathcal{A}_1$ (see Section 3.2 in Zygalakis (2011)) that involves taking the expectation of the Taylor expansion with respect to h of the function $\phi(\bar{U}_{2h}^{(f)}, \bar{X}_{2h}^{(f)}, \bar{u}_{2h}^{(c)}, \bar{x}_{2h}^{(c)})$, for some test function ϕ. $\bar{U}_{2h}^{(f)}, \bar{X}_{2h}^{(f)}$ are the approximations of $\bar{U}(2h), \bar{X}(2h)$ after two fine steps of size h and $\bar{u}_{2h}^{(c)}, \bar{x}_{2h}^{(c)}$ are the approximations of $\bar{u}(2h), \bar{x}(2h)$ after one coarse step of size $2h$. Their form depends on the timestepping method and this is the output of the classes TimestepSymplecticEuler(·), TimestepGeometricLangevinEuler(·) and TimestepBAOAB(·) as can be seen in the code. The function A1(·) then takes these approximations and produces the Taylor expansion.

The third step after computing the generator $\mathcal{L}_0$ and operator $\mathcal{A}_1$ is to compute $\mathcal{A}_1\phi - \frac{1}{2}\mathcal{L}_0^2\phi = \mathcal{A}_1\phi - \frac{1}{2}\mathcal{L}_0(\mathcal{L}_0\phi)$ (see equation (3.9) in Zygalakis (2011)) and this is done by the function L1(·) of our Python code.

If we consider the generator $\mathcal{L}_0^h$ of the modified SDEs (5.66)-(5.69) (the equivalent of $\mathcal{L}_0$ for (5.62)-(5.65)) we can write (Zygalakis, 2011)

$$\mathcal{L}_0^h = \mathcal{L}_0 + h\mathcal{L}_1 + h^2\mathcal{L}_2, \tag{5.71}$$

for some generators $\mathcal{L}_1, \mathcal{L}_2$ and with $\mathcal{L}_0$ as in Definition 5.4. We then have that $\mathcal{L}_1\phi = \mathcal{A}_1\phi - \frac{1}{2}\mathcal{L}_0^2\phi$ and the comparison between $\mathcal{L}_1\phi$ and $\mathcal{A}_1\phi - \frac{1}{2}\mathcal{L}_0^2\phi$ gives the modified equations coefficients which are then extracted to a LaTeX file.

For Symplectic Euler, the modified equations coefficients are given by

$$\begin{aligned}v_1^{(\bar{U})}(\bar{X},\bar{U}) = &\frac{\bar{U}^2}{4}\frac{d\bar{\lambda}}{dX}(\bar{X}) - \frac{\bar{U}}{4}\bar{\lambda}^2(\bar{X}) - \frac{\bar{U}}{4}\bar{\lambda}(\bar{X})\frac{\partial^2 \bar{V}}{\partial U \partial X}(\bar{X},\bar{U}) + \frac{\bar{U}}{4}\frac{\partial^2 \bar{V}}{\partial X^2}(\bar{X},\bar{U})\\ &-\frac{1}{4}\frac{\partial \bar{V}}{\partial X}(\bar{X},\bar{U})\bar{\lambda}(\bar{X}) - \frac{1}{4}\frac{\partial \bar{V}}{\partial X}(\bar{X},\bar{U})\frac{\partial^2 \bar{V}}{\partial U \partial X}(\bar{X},\bar{U})\end{aligned}$$

$$
\begin{aligned}
&+\frac{1}{8}\bar{\sigma}^2(\bar{X})\frac{\partial^3\bar{V}}{\partial U^2\partial X}(\bar{X},\bar{U}),\\
v_1^{(\bar{X})}(\bar{X},\bar{U}) = &-\frac{\bar{U}}{4}\bar{\lambda}(\bar{X})-\frac{1}{4}\frac{\partial\bar{V}}{\partial X}(\bar{X},\bar{U}),\\
\Sigma_1^{(\bar{U})}(\bar{X},\bar{U}) = &-\frac{\bar{U}}{4}\frac{d\bar{\sigma}}{dX}(\bar{X})+\frac{1}{4}\bar{\lambda}(\bar{X})\bar{\sigma}(\bar{X})+\frac{1}{4}\bar{\sigma}(\bar{X})\frac{\partial^2\bar{V}}{\partial U\partial X}(\bar{X},\bar{U}),\\
\Sigma_1^{(\bar{X})}(\bar{X},\bar{U}) = &\frac{1}{4}\bar{\sigma}(\bar{X}),\\
v_2^{(\bar{u})}(\bar{x},\bar{u}) = &\frac{\bar{u}^2}{2}\frac{d\bar{\lambda}}{dx}(\bar{x})-\frac{\bar{u}}{2}\bar{\lambda}^2(\bar{x})-\frac{\bar{u}}{2}\bar{\lambda}(\bar{x})\frac{\partial^2\bar{V}}{\partial u\partial x}(\bar{x},\bar{u})+\frac{\bar{u}}{2}\frac{\partial^2\bar{V}}{\partial x^2}(\bar{x},\bar{u})\\
&-\frac{1}{2}\frac{\partial\bar{V}}{\partial x}(\bar{x},\bar{u})\bar{\lambda}(\bar{x})-\frac{1}{2}\frac{\partial\bar{V}}{\partial x}(\bar{x},\bar{u})\frac{\partial^2\bar{V}}{\partial u\partial x}(\bar{x},\bar{u})+\frac{1}{4}\bar{\sigma}^2(\bar{x})\frac{\partial^3\bar{V}}{\partial u^2\partial x}(\bar{x},\bar{u}),\\
v_2^{(\bar{x})}(\bar{x},\bar{u}) = &-\frac{\bar{u}}{2}\bar{\lambda}(\bar{x})-\frac{1}{2}\frac{\partial\bar{V}}{\partial x}(\bar{x},\bar{u}),\\
\Sigma_2^{(\bar{u})}(\bar{x},\bar{u}) = &-\frac{\bar{u}}{2}\frac{d\bar{\sigma}}{dx}(\bar{x})+\frac{1}{2}\bar{\lambda}(\bar{x})\bar{\sigma}(\bar{x})+\frac{1}{2}\bar{\sigma}(\bar{x})\frac{\partial^2\bar{V}}{\partial u\partial x}(\bar{x},\bar{u}),\\
\Sigma_2^{(\bar{x})}(\bar{x},\bar{u}) = &\frac{1}{2}\bar{\sigma}(\bar{x}).
\end{aligned}
$$

With the computation of the modified equations we now have that for the Symplectic Euler method the first two conditions from Theorem 5.3 hold. Also, with the assumptions made on the coefficients of the original model we have that the newly computed coefficients of the modified equations are Lipschitz continuous which means that the final condition from Theorem 5.3 holds. Therefore, we can apply Theorem 3.1 to the extended inhomogeneous model with the Symplectic Euler timestepping method and expect to see a $\mathcal{O}(\epsilon^{-2})$ cost rate for MLMC.

For the Geometric Langevin method we obtain similar modified equations coefficients given by

$$
\begin{aligned}
v_1^{(\bar{U})}(\bar{X},\bar{U}) = &\frac{\bar{U}^2}{4}\frac{d\bar{\lambda}}{dX}(\bar{X})+\frac{\bar{U}}{4}\bar{\lambda}(\bar{X})\frac{\partial^2\bar{V}}{\partial U\partial X}(\bar{X},\bar{U})+\frac{\bar{U}}{4}\frac{\partial^2\bar{V}}{\partial X^2}(\bar{X},\bar{U})\\
&-\frac{1}{4}\frac{\partial\bar{V}}{\partial X}(\bar{X},\bar{U})\bar{\lambda}(\bar{X})-\frac{1}{4}\frac{\partial\bar{V}}{\partial X}(\bar{X},\bar{U})\frac{\partial^2\bar{V}}{\partial U\partial X}(\bar{X},\bar{U})\\
&-\frac{1}{8}\bar{\sigma}^2(\bar{X})\frac{\partial^3\bar{V}}{\partial U^2\partial X}(\bar{X},\bar{U}),\\
v_1^{(\bar{X})}(\bar{X},\bar{U}) = &-\frac{\bar{U}}{4}\bar{\lambda}(\bar{X})-\frac{1}{4}\frac{\partial\bar{V}}{\partial X}(\bar{X},\bar{U}),\\
\Sigma_1^{(\bar{U})}(\bar{X},\bar{U}) = &-\frac{\bar{U}}{4}\frac{d\bar{\sigma}}{dX}(\bar{X})-\frac{1}{4}\bar{\sigma}(\bar{X})\frac{\partial^2\bar{V}}{\partial U\partial X}(\bar{X},\bar{U}),
\end{aligned}
$$

$$
\begin{aligned}
\Sigma_1^{(\bar{X})}(\bar{X},\bar{U}) =&\ \frac{1}{4}\bar{\sigma}(\bar{X}),\\
v_2^{(\bar{u})}(\bar{x},\bar{u}) =&\ \frac{\bar{u}^2}{2}\frac{d\bar{\lambda}}{dx}(\bar{x}) + \frac{\bar{u}}{2}\bar{\lambda}(\bar{x})\frac{\partial^2\bar{V}}{\partial u\partial x}(\bar{x},\bar{u}) + \frac{\bar{u}}{2}\frac{\partial^2\bar{V}}{\partial x^2}(\bar{x},\bar{u}) - \frac{1}{2}\frac{\partial\bar{V}}{\partial x}(\bar{x},\bar{u})\bar{\lambda}(\bar{x})\\
&- \frac{1}{2}\frac{\partial\bar{V}}{\partial x}(\bar{x},\bar{u})\frac{\partial^2\bar{V}}{\partial u\partial x}(\bar{x},\bar{u}) - \frac{1}{4}\bar{\sigma}^2(\bar{x})\frac{\partial^3\bar{V}}{\partial u^2\partial x}(\bar{x},\bar{u}),\\
v_2^{(\bar{x})}(\bar{x},\bar{u}) =&\ -\frac{\bar{u}}{2}\bar{\lambda}(\bar{x}) - \frac{1}{2}\frac{\partial\bar{V}}{\partial x}(\bar{x},\bar{u}),\\
\Sigma_2^{(\bar{u})}(\bar{x},\bar{u}) =&\ -\frac{\bar{u}}{2}\frac{d\bar{\sigma}}{dx}(\bar{x}) - \frac{1}{2}\bar{\sigma}(\bar{x})\frac{\partial^2\bar{V}}{\partial u\partial x}(\bar{x},\bar{u}),\\
\Sigma_2^{(\bar{x})}(\bar{x},\bar{u}) =&\ \frac{1}{2}\bar{\sigma}(\bar{x}).
\end{aligned}
$$

Finally, for BAOAB the modified equations coefficients have much fewer terms and these are given by

$$
\begin{aligned}
v_1^{(\bar{U})}(\bar{X},\bar{U}) &= -\frac{1}{8}\frac{\partial\bar{V}}{\partial X}(\bar{X},\bar{U})\frac{\partial^2\bar{V}}{\partial U\partial X}(\bar{X},\bar{U}),\\
v_1^{(\bar{X})}(\bar{X},\bar{U}) &= 0,\\
\Sigma_1^{(\bar{U})}(\bar{X},\bar{U}) &= 0,\\
\Sigma_1^{(\bar{X})}(\bar{X},\bar{U}) &= 0,\\
v_2^{(\bar{u})}(\bar{x},\bar{u}) &= -\frac{1}{4}\frac{\partial\bar{V}}{\partial x}(\bar{x},\bar{u})\frac{\partial^2\bar{V}}{\partial u\partial x}(\bar{x},\bar{u}),\\
v_2^{(\bar{x})}(\bar{x},\bar{u}) &= 0,\\
\Sigma_2^{(\bar{u})}(\bar{x},\bar{u}) &= 0,\\
\Sigma_2^{(\bar{x})}(\bar{x},\bar{u}) &= 0.
\end{aligned}
$$

As an aside, note that for the Euler-Maruyama method and when $\sigma(X)$ is not constant, it is not possible to compute modified equations as shown in Zygalakis (2011).

Although we expect to see $\mathcal{O}(\epsilon^{-2})$ cost rates for MLMC for all three timestepping methods, there is something important to note about BAOAB. For this method, almost all correction terms are zero and adding just one term in the original SDE gives the modified equations. This implies that we expect to see a smaller bias error for a fixed timestep size when compared to Symplectic Euler and Geometric Langevin since the BAOAB method is just one term away from being a second order method while formally is still first order. This is an important property that we study further in Chapter 6 where we observe in practice a much better performance. Most importantly, the

correction terms of BAOAB are independent of λ which diverges at the lower boundary. This is not true for the other two methods. This means that when λ is large the correction terms of Symplectic Euler and Geometric Langevin are not of order h and therefore a smaller timestep is required (which increases the cost) in order to have a second order weak error for the modified equations.

As we will see in Chapter 6, Theorem 3.1 still holds numerically without using any special regularisation or transformation on the SDE coefficients. Therefore, for practical applications is not necessary to apply any of them on the SDE coefficients. All the numerical results presented later are about the original model with the special boundary conditions treatment (Section 5.2) and the Met Office's constant regularisation (equations (2.24) and (2.25)). All results from Section 5.3 can then be used to provide a better understanding about the theoretical background of the problem.

5.4 Smoothing polynomials for the concentration problem

The last topic that we study in this chapter is the particle concentration. In Section 2.4 we defined the indicator function which can be used as a quantity of interest to estimate the concentration of particles which land in an interval $[a, b]$ after some time T. Mathematically it can be written as

$$\mathbb{E}\left[\mathbb{1}_{[a,b]}(X(T))\right]. \tag{5.72}$$

This is a very important quantity and one of the main interests in atmospheric dispersion modelling as it measures the probability that a particle ends up in a certain region. Therefore, by knowing how many particles are released from the source enables us to estimate how many of them will end up in each region after time T. At the Met Office for example, they might be interested in the ash particle concentration after a volcanic eruption which will determine whether it is safe or not for airplanes to travel.

The discontinuity of the indicator function at the boundaries of the interval $[a, b]$ causes an increase in the variance which reduces the efficiency of the MLMC method. In Figure 5-7 we plot the variance rates for the Geometric Langevin method which are linear and therefore condition (iii) of Theorem 3.1 does not hold. As a result, we do not expect to see the improved cost rates of $\mathcal{O}(\epsilon^{-2})$ when we run the MLMC method.

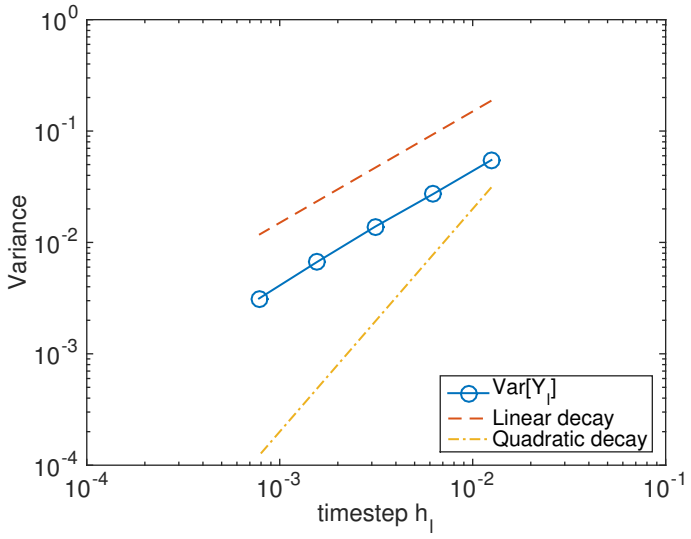


Figure 5-7: Geometric Langevin - Indicator Function

We pick $a = 0.1055$, $b = 0.1555$, $T = 1$, $U_0 = 0.1$ and $X_0 = 0.05$.

Since the indicator function does not produce the optimal rates it becomes necessary to find a suitable approximation. From Theorem 5.3, Lipschitz continuity is a sufficient condition for the modified equations analysis to hold, so this property motivates our choice of approximation. In the next section we follow the method from Giles et al. (2015) and we approximate the indicator function using polynomials which are Lipschitz continuous on bounded domains. Also, we include some analysis showing that the additional errors introduced by this approximation can be minimised. Finally, it is good to note that there might be other methods and possibly more accurate to approximate the indicator function. In this thesis however, we study only the method from Giles et al. (2015) since it helps in the application of the modified equations analysis and improves the performance in practice for MLMC.

5.4.1 Smoothing polynomials

In order to recover the quadratic rates and satisfy condition (iii) of Theorem 3.1, we need to find a Lipschitz continuous function that approximates the indicator function and therefore improves the MLMC variance. One way of doing this is to approximate the indicator function by a polynomial as described in Giles et al. (2015).

Firstly, we can write by linearity

$$\mathbb{E}\left[\mathbb{1}_{[a,b]}(X(T))\right] = \mathbb{E}\left[\mathbb{1}_{(-\infty,b]}(X(T))\right] - \mathbb{E}\left[\mathbb{1}_{(-\infty,a]}(X(T))\right]. \tag{5.73}$$

Then, let us define

$$g_r(x) = \begin{cases} 1 & \text{if} \quad x < -1, \\ p_r(x) & \text{if} \quad -1 \leq x \leq 1, \\ 0 & \text{if} \quad x > 1, \end{cases}$$

where p_r is a polynomial of degree at most $r+1$ which is computed using $p_r(-1) = 1$, $p_r(1) = 0$ and $\int_{-1}^{1} s^j p_r(s) ds = (-1)^j/(j+1), j = 0, \ldots, r-1$. The integer r is chosen by assuming that the density of X is r times continuously differentiable on an open set containing $[S_0, S_1]$ for some $S_0, S_1 \in \mathbb{R}$ with $a, b \in [S_0, S_1]$. The integral equation for computing the polynomial $p_r(s)$ comes from the fact that the function $g_r(x)$ needs to satisfy the property

$$\int_{-1}^{1} s^j \left(\mathbb{1}_{(-\infty,0]}(s) - g_r(s)\right) ds = 0 \text{ for } j = 0, \ldots, r-1, \tag{5.74}$$

as explained in Giles et al. (2015) (which essentiallly means that the first $r-1$ moments of the original density function agree with the corresponding moments of the approximation).

Choosing g_r as above we have the following lemma from Giles et al. (2015).

Lemma 5.5. *There exists a constant $c > 0$ such that for all δ sufficiently small*

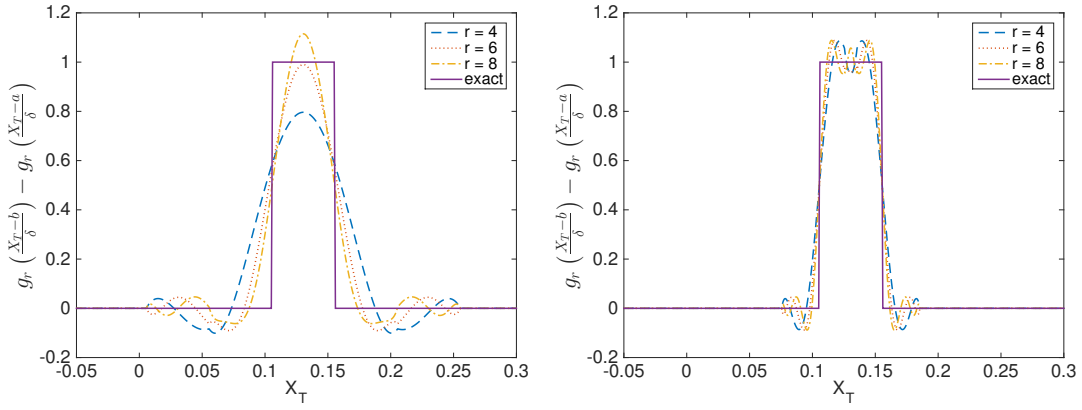
$$\sup_{s \in [S_0, S_1]} \left| \mathbb{E}\left[\mathbb{1}_{(-\infty,s]}(X(T))\right] - \mathbb{E}\left[g_r\left(\frac{X(T)-s}{\delta}\right)\right] \right| \leq c\delta^{r+1}. \tag{5.75}$$

With a suitable choice of δ and r we can approximate any quantity of interest based on the indicator function. We approximate $\mathbb{1}_{[a,b]}(X(T))$ by

$$P^{r,\delta}_{[a,b]}(X(T)) = g_r\left(\frac{X(T)-b}{\delta}\right) - g_r\left(\frac{X(T)-a}{\delta}\right). \tag{5.76}$$

In figures 5-8a and 5-8b we plot the functions $P^{r,\delta}_{[a,b]}(X(T))$ for $\delta = 0.1$ and $\delta = 0.03$ respectively to see graphically how the approximation is affected by the choice of δ and r. As δ decreases and r increases the approximation tends to the indicator

function.



(a) $\delta = 0.1$ (b) $\delta = 0.03$

Figure 5-8: Smoothing polynomials for different values of r, δ and for $a = 0.1055$, $b = 0.1555$.

Since the function $P^{r,\delta}_{[a,b]}$ is now Lipschitz continuous, we expect to recover quadratic variance decay rates in MLMC which implies that the cost will improve asymptotically. However, as $\delta \to 0$ or $r \to \infty$, the smooth approximation tends to the indicator function so we expect the variance rates to drop. We demonstrate this numerically in Section 6.2.1.

5.4.2 Error analysis

We now analyse how the mean and mean square errors of the problem are affected when we approximate the indicator function with a smoothing polynomial. Let $P(X) = \mathbb{1}_{[a,b]}(X)$ and $P^{r,\delta}(X) = P^{r,\delta}_{[a,b]}(X)$ and denote by $\hat{P}_L$ and $\hat{P}^{r,\delta}_L$ the Monte Carlo estimates of $\mathbb{E}[P]$ and $\mathbb{E}[P^{r,\delta}]$ respectively, obtained with a timestep size $h_L = \frac{T}{M_0 2^L}$ (the result holds verbatim for MLMC if h_L is the finest timestep size). Also, let $\mathbb{E}[\hat{P}_L] = \mathbb{E}[P_L]$ and $\mathbb{E}[\hat{P}^{r,\delta}_L] = \mathbb{E}[P^{r,\delta}_L]$. In the next lemma we obtain bounds on the weak and strong errors.

Lemma 5.6. *With the smoothing polynomial technique described in Giles et al. (2015) the mean error can be bounded by*

$$\left|\mathbb{E}[P] - \mathbb{E}[\hat{P}^{r,\delta}_L]\right| \leq 2c\delta^{r+1} + \left|\mathbb{E}[P^{r,\delta}] - \mathbb{E}[P^{r,\delta}_L]\right|, \tag{5.77}$$

and the mean square error by

$$\begin{aligned}\mathbb{E}\left[\left(\hat{P}_L^{r,\delta}-\mathbb{E}[P]\right)^2\right] \leq\ & Var(\hat{P}_L^{r,\delta})+\left(\mathbb{E}[P_L^{r,\delta}]-\mathbb{E}[P^{r,\delta}]\right)^2 \\ & +4c^2\delta^{2(r+1)} \\ & +4c\delta^{r+1}\left|\mathbb{E}[P_L^{r,\delta}]-\mathbb{E}[P^{r,\delta}]\right|, \end{aligned} \tag{5.78}$$

where δ and r are as defined in Section 5.4.1 and c is the constant in (5.75).

$Proof$. We firstly consider the following difference between the expected values of the indicator function P and the smoothed function $P^{r,\delta}$

$$\begin{aligned}\left|\mathbb{E}[P]-\mathbb{E}[P^{r,\delta}]\right| \leq\ & \left|\mathbb{E}\left[\mathbb{1}_{(-\infty,b]}(X_T)\right]-\mathbb{E}\left[g_r\left(\frac{X_T-b}{\delta}\right)\right]\right| \\ & +\left|\mathbb{E}\left[\mathbb{1}_{(-\infty,a]}(X_T)\right]-\mathbb{E}\left[g_r\left(\frac{X_T-a}{\delta}\right)\right]\right| \\ \leq\ & c\delta^{r+1}+c\delta^{r+1} \\ =\ & 2c\delta^{r+1}. \end{aligned} \tag{5.79}$$

Using this expression and the unbiasedness of the Monte Carlo estimator $\hat{P}_L^{r,\delta}$, the mean error bound is

$$\begin{aligned}\left|\mathbb{E}[P]-\mathbb{E}[\hat{P}_L^{r,\delta}]\right| &\leq \left|\mathbb{E}[P]-\mathbb{E}[P^{r,\delta}]\right|+\left|\mathbb{E}[P^{r,\delta}]-\mathbb{E}[\hat{P}_L^{r,\delta}]\right| \\ &\leq 2c\delta^{r+1}+\left|\mathbb{E}[P^{r,\delta}]-\mathbb{E}[P_L^{r,\delta}]\right|. \end{aligned}$$

For the mean square error we write

$$\hat{P}_L^{r,\delta}-\mathbb{E}[P]=\left(\hat{P}_L^{r,\delta}-\mathbb{E}[P_L^{r,\delta}]\right)+\left(\mathbb{E}[P_L^{r,\delta}]-\mathbb{E}[P^{r,\delta}]\right)+\left(\mathbb{E}[P^{r,\delta}]-\mathbb{E}[P]\right),$$

and then we obtain using inequality (5.79)

$$\begin{aligned}\mathbb{E}\left[\left(\hat{P}_L^{r,\delta}-\mathbb{E}[P]\right)^2\right] =\ & \text{Var}(\hat{P}_L^{r,\delta})+\left(\mathbb{E}[P_L^{r,\delta}]-\mathbb{E}[P^{r,\delta}]\right)^2+\left(\mathbb{E}[P^{r,\delta}]-\mathbb{E}[P]\right)^2 \\ & +2\left(\mathbb{E}[P_L^{r,\delta}]-\mathbb{E}[P^{r,\delta}]\right)\left(\mathbb{E}[P^{r,\delta}]-\mathbb{E}[P]\right]) \\ \leq\ & \text{Var}(\hat{P}_L^{r,\delta})+\left(\mathbb{E}[P_L^{r,\delta}]-\mathbb{E}[P^{r,\delta}]\right)^2+4c^2\delta^{2(r+1)} \end{aligned}$$

$$+ 4c\delta^{r+1} \left| \mathbb{E}[P_L^{r,\delta}] - \mathbb{E}[P^{r,\delta}] \right|. \tag{5.80}$$

□

From the mean square error bound we observe that if δ is small enough and r is large enough, the last two terms become small and the error is dominated by the first two terms which represent the sampling error and the discretisation error as in the case of the indicator function without smoothing (see equations (3.18) and (3.27)). Therefore, we can reduce any additional smoothing errors by choosing suitable δ and r. At the same time however we need to be careful not to choose a very small δ or a very large r since the smoothing polynomial approximation will be closer to the indicator function which will cause an increase in the MLMC variance. The effect of δ and r on the MLMC variance is studied numerically in Section 6.2.1.

Chapter 6

Numerical Results

After our theoretical study of the one-dimensional models in Chapters 4 and 5 we now present some numerical results that will enable us to compare the efficiency of Standard and Multilevel Monte Carlo with our three timestepping methods. We include results only for the one-dimensional inhomogeneous model (equations (2.7) and (2.8)) with the Met Office's regularisation as described in Section 2.2 and the special boundary conditions treatment as described in Section 5.2. To be more precise, we do not apply any special regularisation or transformation in the SDE coefficients from those described in Chapters 4 and 5 and these ideas are only used for the theoretical results. As we see, Theorem 3.1 and the convergence of the timestepping methods still hold numerically without the theoretical assumptions on the SDE coefficients.

In Sections 6.1, 6.2 and 6.3 we study the mean particle position, concentration in an interval $[a, b]$ and a piecewise constant approximation of the concentration field respectively at the final time T. We use the three timestepping methods described in Section 3.1 and a uniform timestep. In Section 6.4 we also investigate the effect of adaptive timestepping.

The results are generated using the C++ code initially developed for the paper Müller et al. (2015) and which we later enriched for the purposes of this thesis. The code can be downloaded freely under the GPL license from `https://bitbucket.org/em459/mlmclangevin`. The numerical results and conclusions of this chapter can also be found in Katsiolides et al. (2018) and were obtained with the version which has been archived as Mueller et al. (2017). All plots in this chapter are obtained from Katsiolides et al. (2018) and we follow the same structure as in sections 6 and 7 from

that paper.

We use the model parameters $\kappa_\sigma = 1.3, \kappa_\tau = 0.5$, friction velocity $u_* = 0.2$ and boundary layer height $H = 1$. We also use $T = 1$ and unless otherwise stated the regularisation height is $\epsilon_{\text{reg}} = 0.01$ as discussed in Section 2.2. In Section 6.4.1 we study the impact that ϵ_{reg} has on the numerical results. The initial conditions are $X(0) = 0.05$ and $U(0) = 0.1$. In Section 6.3 we study the effect of the release height on performance.

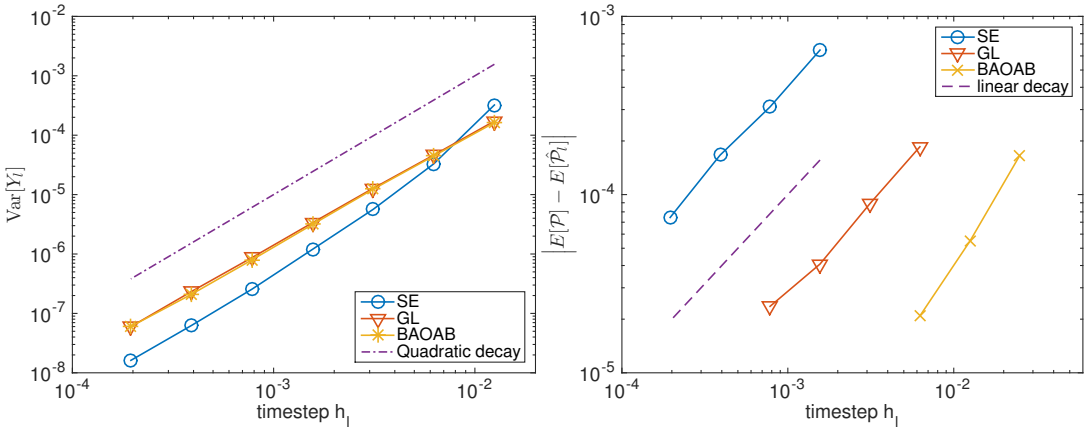
All numerical results are dimensionless and represent units of the reference distance X^{ref}, reference velocity U^{ref} and reference time $t^{\text{ref}} = X^{\text{ref}}/U^{\text{ref}}$. In this chapter (see also Section 2.1.2) we use $X^{\text{ref}} = 10^3 m$, $U^{\text{ref}} = 1m/s$ and therefore $t^{\text{ref}} = 10^3 s \approx 17$ minutes. As a result, the particles are released with an initial upward velocity equal to $0.1m/s$ at a height of $50m$ above ground and results are studied after approximately 17 minutes of travel. Throughout this chapter we measure the computational cost in terms of CPU times in seconds.

6.1 Particle Position

The first quantity of interest that we consider is the mean particle position at time $T = 1$, ie $\mathbb{E}(X(T))$. For all methods we use $M_0 = 40$ timesteps on the coarsest level.

In Figure 6-1 we firstly confirm numerically that the conditions of Theorem 3.1 hold without the theoretical assumptions used before in the proofs. In Figure 6-1a we plot $Var[Y_l]$ as a function of the timestep size h_l and we observe that for all timestepping methods the rate is quadratic, i.e. proportional to h_l^2, which confirms condition (iii) of Theorem 3.1. Geometric Langevin (GL) and BAOAB methods have similar variance that is larger than Symplectic Euler (SE) variance except for the largest timestep where the asymptotic assumptions do not necessarily hold.

In Figure 6-1b we plot the bias error $\left|\mathbb{E}[\hat{\mathcal{P}}_l - \mathcal{P}]\right|$ as a function of h_l and we observe that the rate is linear, i.e. proportional to h_l. This confirms that condition (i) holds for all timestepping methods and therefore confirms numerically Theorem 3.1. A very important observation is that the bias error of the splitting methods is much smaller when compared to the bias error of the SE method. In particular, GL has about $13\times$ smaller bias error than SE and BAOAB reduces this error by an additional factor of 4. The total error reduction of BAOAB relative to SE is then $52\times$. As discussed in Section 5.3.2, this difference in the bias error was expected since the modified equa-



(a) Variance rates

(b) Bias error

Figure 6-1: Variance and bias error - Mean particle position

tions analysis suggests that BAOAB does not have issues with the large $\lambda(X)$ at the lower boundary and is almost a second order method which means that you can use a larger timestep size in order to achieve a certain tolerance.

In Figure 6-2 we plot the total computational costs of the StMC and MLMC methods for all discretisation schemes. The cost is plotted as a function of the root mean square error tolerance ϵ. The StMC cost rates are asymptotically proportional to ϵ^{-3} for all timestepping methods, as discussed in Section 3.2.1. The MLMC cost rates are asymptotically proportional to ϵ^{-2} as expected from Theorem 3.1.

Although the cost of StMC and MLMC methods show the expected dependency on ϵ, there are significant differences in the absolute performance for fixed ϵ. Comparing only the MLMC cost rates we can see that all timestepping methods perform similarly. They have some small differences and SE is slightly cheaper. However, performance between different time integrators varies very strongly if we compare the different StMC methods among themselves. In particular, the GL method is approximately $5\times$ faster than SE and BAOAB is more than $20\times$ faster. The main reason for this observation is the large difference in the bias error as discussed for the results presented in Figure 6-1b. The splitting methods have a smaller bias, which allows the use of a larger timestep at fixed ϵ, resulting in an overall cheaper method.

Next, we briefly discuss the timestep sizes and number of samples used to generate Figure 6-2. In Table 6.1 we give the maximum and minimum timestep size used for each timestepping method with the corresponding ϵ in the brackets. For ev-

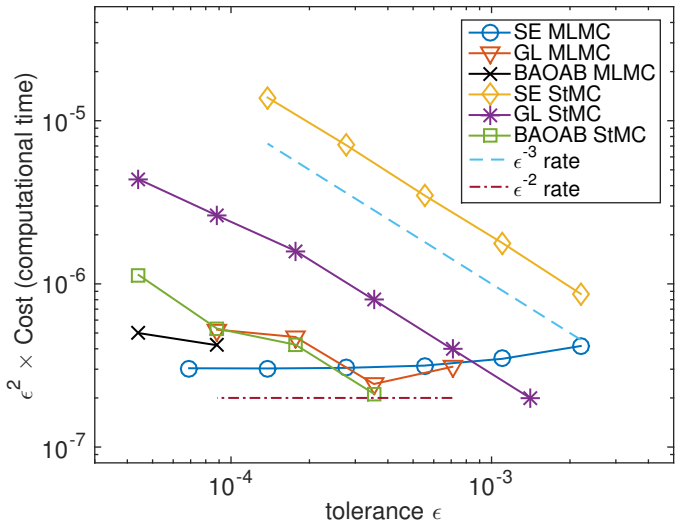


Figure 6-2: Computational cost for varying tolerance - Mean particle position

	SE	GL	BAOAB
maximum	$h_L = 3.1 \cdot 10^{-3}$ $(\epsilon = 2.2 \cdot 10^{-3})$	$h_L = 2.5 \cdot 10^{-2}$ $(\epsilon = 1.4 \cdot 10^{-3})$	$h_L = 2.5 \cdot 10^{-2}$ $(\epsilon = 3.5 \cdot 10^{-4})$
minimum	$h_L = 9.8 \cdot 10^{-5}$ $(\epsilon = 6.9 \cdot 10^{-5})$	$h_L = 7.8 \cdot 10^{-4}$ $(\epsilon = 4.4 \cdot 10^{-5})$	$h_L = 3.1 \cdot 10^{-3}$ $(\epsilon = 4.4 \cdot 10^{-5})$

Table 6.1: Range of timestep sizes used for each timestepping method with their corresponding ϵ.

ery timestepping method the StMC timestep size and the MLMC timestep size on the finest level is the same for a given root mean square error ϵ (due to the telescoping sum equation (3.22)). In Appendix A.4 we include some details on the number of samples N_l on each level l for two MLMC runs.

Our overall conclusion is that for small tolerances MLMC, with any timestepping method can improve the efficiency, but for larger tolerances ($> 5 \cdot 10^{-4}$) the StMC method with BAOAB is the most efficient.

6.2 Concentration

The second quantity of interest that we consider is the particle concentration in an interval $[a, b] \subset [0, 1]$ at the final time T; we use $a = 0.1055$ and $b = 0.1555$ here. The particle concentration can be written as $\mathbb{E}\left[\mathbb{1}_{[a,b]}(X(T))\right]$ and we replace the indi-

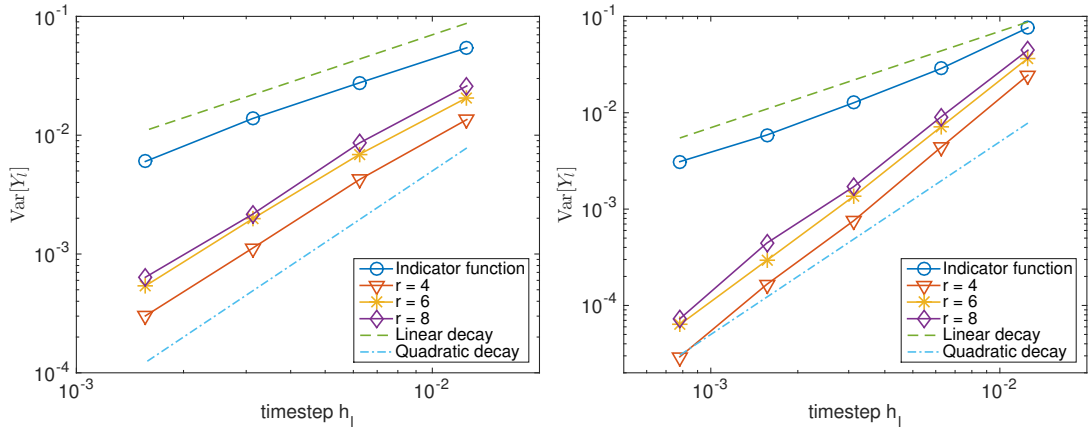
cator function with a smoothing polynomial $P^{r,\delta}_{[a,b]}(X(T))$ as discussed in Section 5.4. Firstly, we check the sensitivity on the smoothing parameters δ and r and then present some similar experiments as for the mean particle position case.

6.2.1 Sensitivity to smoothing parameters

As discussed in Section 5.4, using the indicator function as a quantity of interest causes a decrease in the MLMC variance rate so we replace it with the smoothing polynomial $P^{r,\delta}_{[a,b]}(X(T))$ defined in equation (5.76). In Figures 6-3a and 6-3b we firstly plot $Var[Y_l]$ as a function of the timestep size h_l for GL and SE methods to verify condition (iii) of Theorem 3.1. In both plots we use $\delta = 0.1$, three different values of r and the quantity of interest is $P^{r,\delta}_{[a,b]}(X(T))$. We also plot results for the indicator function $\mathbb{1}_{[a,b]}$ to see how the variances compare. In all cases the tolerance on the root mean square is $\epsilon = 10^{-3}$ and since the SE method has a larger bias error compared to GL we must use a smaller finest timestep size. We do not include any results for BAOAB since they are very similar to those of GL. We observe that for the smoothed polynomials the variance decays with a quadratic rate and for the non-smooth indicator function the rate is linear as in Figure 5-7. Therefore, Theorem 3.1 holds numerically when using any of the smoothed polynomials and the cost rate is expected to improve asymptotically. Note however that we do not study how the overall cost (i.e. the constants in equations (3.20) and (3.24)) is affected by this change but as we see later the asymptotic cost rate of MLMC will improve. Also, note that the absolute value of the variance grows as r increases which is expected since $P^{r,\delta}_{[a,b]}$ converges to the non-smooth indicator function as $r \to \infty$.

In Section 5.4.2 we presented some analysis regarding the errors introduced by the smoothing polynomial technique. We concluded that as $\delta \to 0$ or $r \to \infty$ the smoothing errors are decreasing.

Since we now have to choose a pair of values of δ and r for our numerical experiments we must show that for those values the additional error which is introduced by replacing the indicator function by $P^{r,\delta}_{[a,b]}(X(T))$ is indeed small. In Table 6.2 (Katsiolides et al., 2018) we present some estimates of the expected values of $\mathbb{1}_{[a,b]}$ and of $P^{r,\delta}_{[a,b]}$ for different values of r and a fixed $\delta = 0.1$. The timestep size h is such that the bias error is less than $\epsilon/\sqrt{2}$ with $\epsilon = 10^{-3}$ and we also choose a large number of samples such that the standard deviation of each estimator is less



(a) Geometric Langevin (b) Symplectic Euler

Figure 6-3: Variance - Concentration $P^{r,\delta}_{[a,b]}$ with $\delta = 0.1$

Quantity of interest	r	value			difference		
		GL	SE	BAOAB	GL	SE	BAOAB
$\mathbb{E}\left[P^{r,\delta}_{[a,b]}(X(T))\right]$	4	0.16678	0.16705	0.16687	$1.2 \cdot 10^{-5}$	$7.4 \cdot 10^{-5}$	$3.1 \cdot 10^{-5}$
	6	0.16669	0.16709	0.16677	$7.2 \cdot 10^{-5}$	$4.1 \cdot 10^{-5}$	$7.5 \cdot 10^{-5}$
	8	0.16682	0.16714	0.16683	$5.1 \cdot 10^{-5}$	$9.9 \cdot 10^{-6}$	$1.6 \cdot 10^{-5}$
$\mathbb{E}\left[\mathbb{1}_{[a,b]}(X(T))\right]$		0.16677	0.16713	0.16684			

Table 6.2: Estimates of $\mathbb{E}\left[P^{r,\delta}_{[a,b]}(X(T))\right]$ and $\mathbb{E}\left[\mathbb{1}_{[a,b]}(X(T))\right]$ for $\delta = 0.1$ and different values of r.

than $3.5 \cdot 10^{-5}$. In the last three columns of Table 6.2 we compute the difference $\left|\mathbb{E}\left[P^{r,\delta}_{[a,b]}(X(T))\right] - \mathbb{E}\left[\mathbb{1}_{[a,b]}(X(T))\right]\right|$ which quantifies the error from the polynomial approximation of the indicator function and we can see that is of the order $5 \cdot 10^{-5}$ in all cases. Therefore, we conclude that when compared to the bias (which is $\sim 10^{-3}$) the smoothing error is clearly negligible for our choices of δ and r.

We also verify that the additional cost of evaluating $P^{r,\delta}_{[a,b]}$ is negligible. It is less than the cost of one timestep with a very small dependence on r. In addition, we have confirmed that the results observed in Table 6.2 are independent of the choice of the interval $[a, b]$ and therefore in what follows we choose $\delta = 0.1$, $r = 4$ and neglect the smoothing error.

The drawback of the above approach is that you need to verify that the smoothing error is small by doing numerical tests at the start as in Table 6.2. In Giles et al. (2017)

this problem is addressed by presenting an algorithm with on-the-fly estimations. In this thesis however we just use the smoothing parameters as above and the reader is referred to Giles et al. (2017) for any further details.

6.2.2 Cost comparison between Standard and Multilevel Monte Carlo

We now compute the cost of StMC and MLMC methods as a function of the tolerance on the root mean square error. For the SE and GL methods we choose $M_0 = 80$ and for BAOAB $M_0 = 40$. We set $\delta = 0.1$ and $r = 4$ as justified by the error estimates from the previous section.

In Figure 6-4 we plot the total computational cost of the Monte Carlo methods for all discretisation schemes. The cost is plotted as a function of the root mean square error tolerance ϵ and we see similar results as in the mean particle position case. The cost rates of the StMC methods are asymptotically proportional to ϵ^{-3} and the cost rates of the MLMC methods are asymptotically proportional to ϵ^{-2} as expected. Comparing only the MLMC results among themselves we can see a similar performance for all timestepping methods. Comparing the StMC results among themselves we again observe a large difference in the performance and BAOAB is the fastest method for the same reason discussed in Section 6.1.

Our conclusion is similar as before. For tight tolerances is better to use MLMC with any timestepping method but for larger tolerances the StMC with a splitting method is cheaper overall. Lastly, in Appendix A.4 we include details for the number of samples on each MLMC level for a representative run where we can see that the largest number of samples is found on the coarsest level which is the cheapest to compute.

6.3 Probability Density Function

Our final quantity of interest is the probability density function (p.d.f.) which is very important in atmospheric dispersion since a piecewise constant approximation of the concentration field is the main output of dispersion models like NAME. For this approximation we divide our space interval $[0, H]$ to a set of smaller intervals at equidistant points $0 = a_0 < a_1 < a_2 < \cdots < a_{k-1} < a_k = H$ and consider the vector-valued

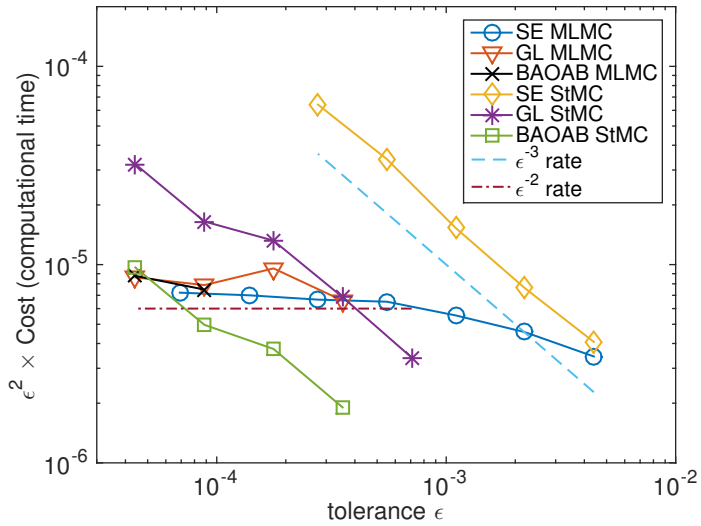


Figure 6-4: Computational cost for varying tolerance - $P_{[a,b]}^{r=4,\delta=0.1}(X_T)$

quantity of interest

$$\mathcal{P} = (\mathcal{P}_0, \mathcal{P}_1, \ldots, \mathcal{P}_k) = \left(P_{[a_0,a_1]}^{r,\delta}(X(T)), P_{[a_1,a_2]}^{r,\delta}(X(T)), \ldots, P_{[a_{k-1},a_k]}^{r,\delta}(X(T)) \right), \tag{6.1}$$

i.e. we calculate the integral of the concentration in each interval $[a_i, a_{i+1}]$. This will then give the required approximation since for the p.d.f. $\rho(X)$ we have

$$\int_a^b \rho(X)dX = \mathbb{E}\left[\mathbb{1}_{[a,b]}\right] \approx \mathbb{E}\left[P_{[a,b]}^{r,\delta}\right] \qquad \forall [a,b] \subset [0,H]. \tag{6.2}$$

The piecewise constant approximation of $\rho(X)$ is then obtained by dividing the quantity of interest in equation (6.2) by the length of the interval. As before we use $\delta = 0.1$ and $r = 4$. As discussed in Section 3.2.2, in order to compute the number of samples N_l on each level of the MLMC algorithm for a scalar-valued quantity of interest, we use the following formula from Giles (2008)

$$N_l = \left\lceil 2\epsilon^{-2}\sqrt{Var(Y_l)h_l}\left(\sum_{i=0}^{L}\sqrt{Var(Y_i)/h_i}\right)\right\rceil, \tag{6.3}$$

where $Var(Y_l)$ is the variance on level l. Since we now have a vector-valued quantity of interest we estimate N_l by choosing the maximal absolute variance over all intervals

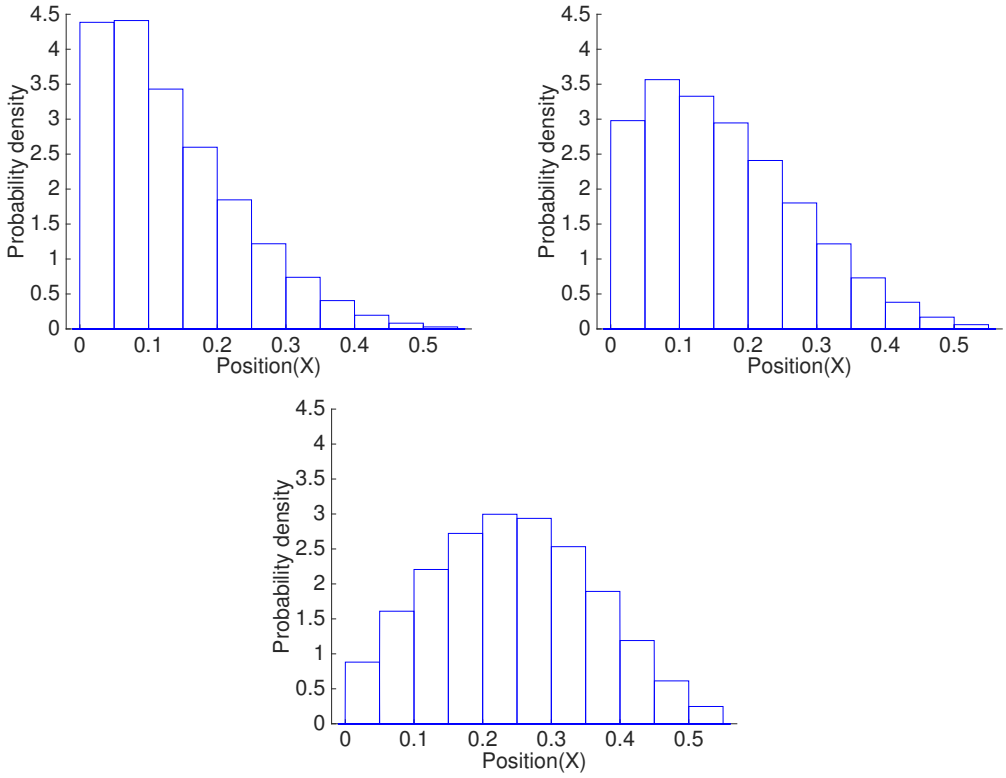


Figure 6-5: Piecewise constant approximation of probability density function for release height 0.05 (top left), 0.10 (top right) and 0.20 (bottom).

(which guarantees that the error is below the tolerance in all intervals). In particular, we let $\mathcal{P}_{l,i} = P^{r,\delta}_{[a_i,a_{i+1}]}(X_{T,l})$ and $Y_{0,i} = \mathcal{P}_{0,i}, Y_{l,i} = \mathcal{P}_{l,i} - \mathcal{P}_{l-1,i}$, for $l = 1, \ldots, L$ and $i = 0, \ldots, k-1$. Then, we use $\max_{i=0,\ldots,k-1} |Var(Y_{l,i})|$ in place of $Var(Y_l)$ in equation 6.3 (Katsiolides et al., 2018).

With the p.d.f. setting as above we can now construct density function plots. In Figure 6-5 we plot piecewise constant approximations of the p.d.f. of $X(T)$ using Symplectic Euler and MLMC with intervals of size 0.05. The plot on the top left uses $X(0) = 0.05$ that corresponds to a release height of $50m$ above ground, which is the same initial condition used to obtain all results in Sections 6.1 and 6.2. We also include results for $X(0) = 0.1$ and $X(0) = 0.2$ since we found that the computational cost is very sensitive to the release height. These heights correspond to $100m$ and $200m$ above ground, respectively. For all release heights the initial velocity is given by $U(0) = 0.1$.

In Table 6.3 we present the computational cost in CPU times necessary to compute

release height	StMC			MLMC	
	SE	GL	BAOAB	SE	GL
0.05 $(50m)$	2561.41	321.77	91.21	218.52	265.84
0.10 $(100m)$	867.32	277.72	79.25	120.17	125.92
0.20 $(200m)$	316.33	280.04	78.15	64.78	103.69

Table 6.3: Computational cost (CPU time in seconds) for different release heights.

the p.d.f. with a root mean square error below a tolerance of $\epsilon = 4 \cdot 10^{-3}$. For SE and GL we use $M_0 = 80$ and for BAOAB we use $M_0 = 40$.

Comparing the StMC results for each release height separately we can see that with GL and BAOAB we have a large increase in the performance when compared to SE. The reason for this is the same as in the previous quantities of interest that we studied and is due to the fact that the splitting methods are less affected by the large values of $\lambda(X)$ near the ground. The bias error is therefore smaller for BAOAB and as a result we can use larger timesteps. As $X(0)$ increases however, the particles spend less time near the ground and the bias error of the SE method decreases which gives an improvement in the performance. GL and BAOAB are not affected much by the release height and there is no significant change in the bias error.

For the MLMC algorithms we can see that the performance improves as the release height increases. Results for BAOAB are not included since the number of levels is too small to achieve any speed-up for the chosen tolerance. Comparing Standard and Multilevel Monte Carlo results we can see that MLMC is faster especially for the low release heights. Finally, for $X(0) = 0.05$ and $X(0) = 0.1$ the fastest method is StMC-BAOAB and for $X(0) = 0.2$ the fastest method is MLMC-SE.

We conclude that the splitting methods and the MLMC estimator significantly reduce the cost of computing p.d.f.'s when compared to the StMC-SE algorithm used in many Lagrangian atmospheric dispersion models. For a low release height StMC-BAOAB leads to speedups of $28\times$ ($X(0) = 0.05$) and $11\times$ ($X(0) = 0.10$) when compared to StMC-SE. At higher release heights ($X(0) = 0.20$) MLMC method with the SE integrator is $5\times$ faster than the algorithm currently used in many operational atmospheric dispersion models.

As a final comment we note that there are also alternative methods to etsimate p.d.f.'s, like for example the kernel density method used in Ramli and Esler (2016). In this thesis however, we only consider the method described above and the reader is

referred to Ramli and Esler (2016) for more details.

6.4 Adaptive timestepping

In Section 3.1 we defined our timestepping methods and we also discussed the stability constraints of the SE method. For a reminder, we deduced the following constraint on the timestep size h

$$h < \frac{2}{\max\limits_{X\in[0,H]} \lambda(X)} = \frac{2}{\lambda(\epsilon_{\text{reg}})}, \tag{6.4}$$

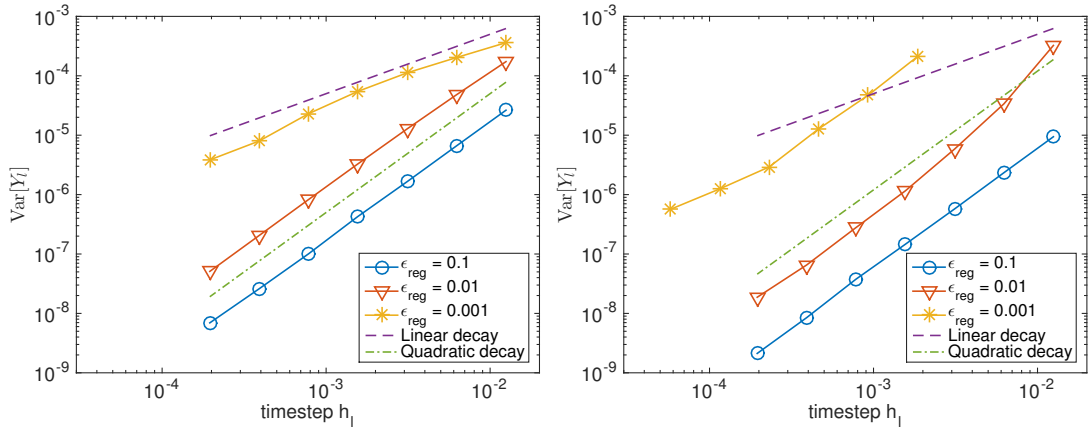
where $\lambda(X)$ is as in equation (2.7). The function $\lambda(X)$ is very large close to the ground, which leads to tight constraints on the timestep size as can be seen from inequality 6.4. However, in the rest of the domain it is unnecessary to have a very small timestep, making the method prohibitively expensive. To address this problem, we adaptively choose smaller timesteps in those parts of the domain where they are required to ensure stability. In the literature there are various methods that describe adaptive timestepping in MLMC and some of them can be found in Hoel et al. (2012), Hoel et al. (2014), Giles et al. (2016). In this thesis we apply the method described in Giles et al. (2016).

6.4.1 Sensitivity to regularisation height

An important question is the sensitivity on the regularisation height ϵ_{reg}. In Figures 6-6a and 6-6b we plot $Var[Y_l]$ as a function of the timestep size h_l to check whether the decay is quadratic as required by condition (iii) of Theorem 3.1. We show results for SE and GL timestepping methods as ϵ_{reg} is varied and in both cases the quantity of interest is the particle position $X(T)$ with the same model parameters used in section 6.1.

For GL we use $M_0 = 40$ timesteps on the coarsest level. For SE we must be more careful with the timestep sizes since we have the stability constraint described by equation (6.4). In addition, $\lambda(\epsilon_{\text{reg}})$ increases with decreasing ϵ_{reg} which further restricts the timestep size. For $\epsilon_{\text{reg}} = 0.1$ and $\epsilon_{\text{reg}} = 0.01$ we use the same timestep sizes as for the GL method but for $\epsilon_{\text{reg}} = 0.001$ we use more timesteps on the coarsest level by setting $M_0 = 270$.

From the plots we observe that for $\epsilon_{\text{reg}} = 0.1$ and $\epsilon_{\text{reg}} = 0.01$ the variance rate



(a) Geometric Langevin

(b) Symplectic Euler

Figure 6-6: Variance rates for different ϵ_{reg} - Mean particle position

is quadratic but for $\epsilon_{\text{reg}} = 0.001$ the rate is linear. The reason why the variance is linear for small ϵ_{reg} can be explained using the theory of modified equations (Shardlow, 2006), (Zygalakis, 2011), (Müller et al., 2015) considered in Section 5.3.

As $\epsilon_{\text{reg}} \to 0$ we have that $\max_{X \in [0,H]} \lambda(X) \to \infty$ and therefore the correction terms of the modified equations containing $\lambda(X)$ (see equations (5.66) and (5.67)) are not of order h_l since $h_l \lambda(X)$ is not small enough. The result of this is that the modified equations analysis breaks down. However, as $h_l \to 0$ for fixed ϵ_{reg}, the term $h_l \lambda(X)$ becomes smaller and we recover the expected quadratic variance decay rate since the correction terms are of order h_l again. Also, the value of ϵ_{reg} affects the absolute value of the variance and as we can see from the plots the variance increases with decreasing ϵ_{reg} again due to the fact that $\lambda(X)$ varies strongly near the lower boundary.

For all experiments in the previous sections we used $\epsilon_{\text{reg}} = 0.01$ which is similar to what the Met Office uses (see Section 2.2). The variance rate is quadratic and therefore condition (iii) of Theorem 3.1 is satisfied. In addition, we have seen that by decreasing the timestep size, the variance decreases but at the same time the total cost increases.

6.4.2 Adaptive timestepping

Our adaptive MLMC method is as described in Giles et al. (2016) and we adjust the timestep size at every time t_n such that the method is stable. In particular, we use

$$h_n^{(\text{adaptive})} := \min\left\{h, \frac{\lambda(X_{\text{adapt}})}{\lambda(X_n)}h\right\}, \tag{6.5}$$

where X_n is the current particle position and X_{adapt} is a reference height. With this choice of adaptive timestep we then have

$$h_n^{(\text{adaptive})}\lambda(X_n) < \lambda(X_{\text{adapt}})h, \tag{6.6}$$

so if we pick a value for the maximal timestep h and X_{adapt} such that $\lambda(X_{\text{adapt}})h < 2$ we then guarantee the stability of the SE method (see inequality (6.4)). A very important observation about this method of adaptive timestepping is that it is independent of the MLMC level and therefore the telescoping sum in equation (3.22) is not violated which means that it does not introduce any additional bias (Giles et al., 2016).

Now since $\lambda(X)$ is a decreasing function we only change the timestep size when the particles are below the height X_{adapt}. Note however that although the product $h_n^{(\text{adaptive})}\lambda(X_n)$ is always bounded such that the SE method is stable, we still need to regularise $\lambda(X)$ as in Section 2.2. This is because without regularisation we have $\lambda \to \infty$ near the ground which implies that $h_n^{(\text{adaptive})} \to 0$ and therefore it can take an infinite amount of time to calculate just one trajectory since the number of timesteps might not be bounded..

As we explain below, when using adaptivity, the fine and coarse steps are not necessarily nested as in the uniform timestepping (two fine steps correspond to one coarse step) and it is possible to have more than two steps on the fine level corresponding to one step on the coarse level. Details of the adaptive MLMC algorithm for the Euler method can be found in Giles et al. (2016) and here we only give some details for how to construct the random increments. In addition, we describe how the boundary conditions treatment (Section 5.2) can be combined with adaptivity as presented in Katsiolides et al. (2018).

On the coarse level, the interval $[0, T]$ is partitioned into small sub-intervals through the points $t_i^{(c)}$ and one step of the numerical method calculates the solution at time $t_i^{(c)}$, given the solution at time $t_{i-1}^{(c)}$. Similarly, on the fine level, the interval $[0, T]$ is divided

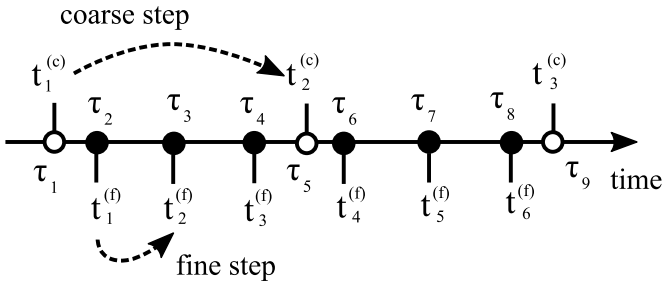


Figure 6-7: Non-nested adaptive timesteps

into sub-intervals by the points $t_i^{(f)}$. However, in contrast to the previously described MLMC method where for each $t_i^{(c)}$ there existed a j such that $t_j^{(f)} = t_i^{(c)}$, the sub-intervals on two subsequent levels are not nested, i.e. the $t_i^{(c)}$ are not related to the $t_i^{(f)}$.

With these two sets of partition points the interval $[0, T]$ is then divided into a number of sub-intervals $[\tau_j, \tau_{j+1}], j = 0, \ldots, N-1$ such that $\tau_0 = 0$, $\tau_N = T$. Each τ_j can be either a fine or a coarse time point as defined above and we denote this graphically in Figure 6-7 (adapted from Figure 1 in Giles et al. (2016)). In particular, we define

$$\tau_j = \begin{cases} t_i^{(c)} & \text{for coarse time points,} \\ t_i^{(f)} & \text{for fine time points,} \end{cases} \tag{6.7}$$

and for each interval $[\tau_j, \tau_{j+1}]$ we generate an independent random variable $\xi_j \sim \mathrm{Normal}(0, 1)$. Then, by using the algorithm from Giles et al. (2016) for SE we can combine these random numbers for each interval $[t_i^{(x)}, t_{i+1}^{(x)}]$ (where $x = c, f$) and construct a random increment $\Delta W(t_i^{(x)}, t_{i+1}^{(x)}) \sim \mathrm{Normal}(0, t_{i+1}^{(x)} - t_i^{(x)})$. If in addition we multiply each generated random variable ξ_j with $S_j^{(\cdot)} = (-1)^{\text{number of reflections up to time } \tau_j}$ as in Section 5.2 we obtain an increment which takes into account both adaptive timestepping and boundary conditions treatment. From Katsiolides et al. (2018) the random increment for the SE coarse step is given by

$$\Delta W(t_i^{(c)}, t_{i+1}^{(c)}) = S_i^{(c)} \sum_{j=j_-}^{j_+-1} S_j^{(f)} \xi_j \sqrt{\tau_{j+1} - \tau_j}, \quad \text{where } \tau_{j_-} = t_i^{(c)} \text{ and } \tau_{j_+} = t_{i+1}^{(c)}, \tag{6.8}$$

and replaces the term $\sqrt{h}\xi_n$ in equation (3.1). It is also easy to see that it has variance $t_{i+1}^{(c)} - t_i^{(c)}$ since it is a sum of Gaussian random variables. For the GL coarse step we

use

$$\Delta W(t_i^{(c)}, t_{i+1}^{(c)}) = S_i^{(c)} \sum_{j=j_-}^{j_+-1} \Bigg\{ S_j^{(f)} \xi_j \sqrt{\frac{1-\exp[-2\lambda(X(\tau_j))\cdot(\tau_{j+1}-\tau_j)]}{2\lambda(X(\tau_j))}} \cdot \exp\left[-\sum_{k=j+1}^{j_+-1} \lambda(X(\tau_k))\cdot(\tau_{k+1}-\tau_k) \right] \Bigg\}, \tag{6.9}$$

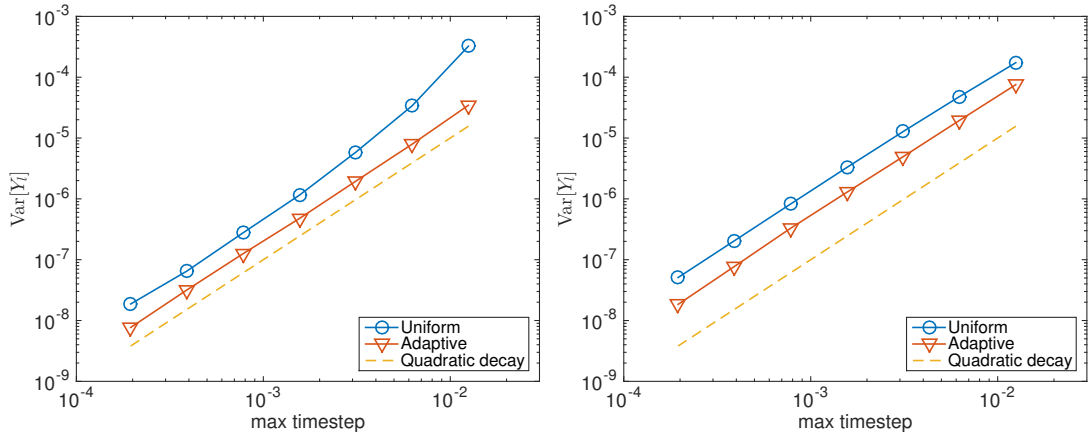
which replaces the term $\sqrt{\frac{1-\exp[-2\lambda(X_n)h]}{2\lambda(X_n)}}\xi_n$ in equation (3.8). For both methods we use the same expression for the fine step random increment $\Delta W(t_i^{(f)}, t_{i+1}^{(f)})$ but without the factors $S_j^{(\cdot)}$.

6.4.3 Numerical results

We now present some numerical results for adaptive timestepping. As a quantity of interest we use the mean particle position with the same model parameters as in Section 6.1 and $X_{\text{adapt}} = 0.05$. We use both SE and GL timestepping methods. In Figure 6-8 we first plot the variance rates to check numerically if condition (iii) from Theorem 3.1 holds, i.e. if we have quadratic variance decay rates when adaptivity is used. We observe that for both methods the variance decays with a quadratic rate with respect to the maximal timestep h. Also, we observe that the absolute value of the variance decreases with adaptive timestepping which is expected since on average we use a smaller timestep size when compared to the uniform case.

In Figure 6-9 we plot the corresponding cost rates. In Figure 6-9a we can see that adaptivity reduces the cost of the SE method. As ϵ becomes smaller however, the two costs converge to the same value since a smaller ϵ implies a smaller timestep and therefore we adapt less times when using a fixed X_{adapt}. Also, the bias error for SE is reduced ($\approx 6\times$) when using adaptivity which is one of the reasons why the cost decreases.

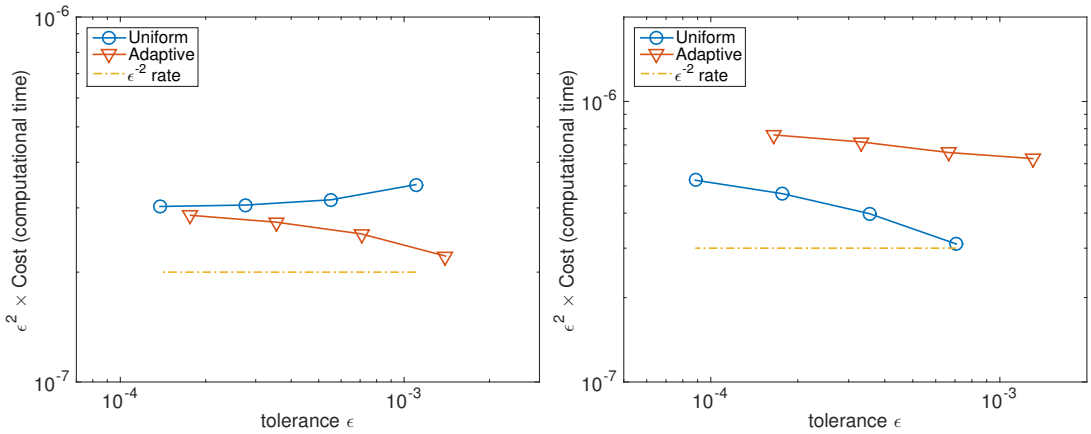
Now by looking at Figure 6-9b and the cost of the GL method we can see that adaptivity increases the cost. The main reason for the increase in the cost is that in this case adaptivity increases the bias error ($\approx 3\times$). This is a counter-intuitive result since with adaptivity a smaller timestep is used on average when comparing to the uniform case. In the ODE case one obviously expects the error to be smaller with adaptive timestepping, but in the SDE case this is less clear. Studying the problem theoretically



(a) Symplectic Euler (b) Geometric Langevin

Figure 6-8: Variance rates - Uniform and adaptive timesteps

to see what behaviour we should expect is hard for our non-linear SDE and beyond the scope of this thesis. However, to make sure that there are no bugs in the code we made several tests like for example testing the case when $\sigma = 0$. For this ODE we confirmed that GL reduces the bias error. Further investigation of this problem which might be linked to the divergence of $\lambda(X)$ near the ground, can be a topic for future research. Also, it will be interesting to see results when using the BAOAB method which can be another direction for future research.



(a) Symplectic Euler (b) Geometric Langevin

Figure 6-9: Cost rates - Uniform and adaptive timesteps

Chapter 7

Higher Dimensional Models

After presenting theoretical and numerical results for the one-dimensional models we now study higher dimensional models which give a better description of the atmospheric conditions since turbulence is not just one-dimensional. In a realistic model for the transport and spread of particles in the atmosphere there are two main things happening: Firstly, the particles are advected by a background velocity field, which at the Met Office is obtained from the numerical forecast model. However, this field has a finite resolution, so can only represent eddies which are larger than some cutoff Λ. Secondly, any processes at scales below this cutoff are modelled by the turbulent component $\mathbf{U}(t)$ of the particles' velocity. In the model, this is described by the random terms.

Based on the above two properties, we construct a set of synthetic background velocity fields and study the transport and spread of particles in those fields. The background velocity field is constructed as a sum of modes with the correct energy spectrum and is such that it is non-divergent in order to avoid the unphysical accumulation of particles in some parts of the domain. The non-divergent property was also one of the reasons that we did not consider a background velocity in the one-dimensional case since this would require a constant vertical quantity in the entire boundary layer which is not physical. Lastly, the parameters of the turbulence model are adjusted such that they correctly represent the unresolved processes at the scales below the cutoff Λ.

Our models have the general form given in the Introduction by equations (1.1) and (1.2) for $d = 2$ and $d = 3$. For $d = 2$ the model will describe horizontal two-dimensional homogeneous turbulence. Since the atmospheric boundary layer is very

thin when you consider a global scale, this model can be a good approximation and therefore we study it first. For $d = 3$ the model will combine the horizontal structure of the homogeneous two-dimensional model with the vertical structure of the inhomogeneous one-dimensional model intoduced in Section 2.1. In Sections 7.1 and 7.2 we study the two- and three-dimensional models respectively and we present some numerical results to compare the efficiency of Standard and Multilevel Monte Carlo with the Symplectic Euler method. Finally, in Section 7.3 we study the effect on the performance when we change some of the model parameters and we present some numerical results with the BAOAB method.

All numerical results in this chapter were generated with an updated version of the C++ code used in Chapter 6. The code can be downloaded freely under the GPL license from `https://bitbucket.org/em459/mlmclangevin`.

7.1 Two-dimensional model for homogeneous turbulence

Our two-dimensional model describes homogeneous turbulence on the horizontal plane and $\mathbf{U}(t) = (U_1(t), U_2(t))$, $\mathbf{X}(t) = (X_1(t), X_2(t))$ denote the two-dimensional velocity and position respectively. The model is given by the following SDEs

$$dU_i(t) = -\frac{U_i(t)}{\tau}dt + \sqrt{\frac{2\sigma_U^2}{\tau}}dW_i(t), \tag{7.1}$$

$$dX_i(t) = (v_i(\mathbf{X}(t), t) + U_i(t))\, dt, \quad i = 1, 2, \tag{7.2}$$

which are a special case of the general model given by equations (1.1) and (1.2) with $d = 2$. $W_1(t)$ and $W_2(t)$ are independent Brownian motions and we assume that there are no boundary conditions so that the particles can travel anywhere.

Since the turbulence is homogeneous, τ (velocity decorrelation timescale) and σ_U^2 (variance at equilibrium) are constants and we also assume that they are isotropic (i.e. the same in all directions). Equation (7.1) can therefore be seen as a higher dimensional extension of the one-dimensional homogeneous model described by equation (2.13). τ and σ_U could be different in each direction and also depend on $\mathbf{X} = (X_1, X_2)$ but for simplicity we do not consider any of these cases. The reader can refer to Webster et al. (2003) for more information on how these functions are chosen for practical

applications. We define τ and σ_U in Section 7.1.2. Also note that in the most general case we could have something like

$$dU_i(t) = -\sum_{j=1}^{d} A_{ij}U_j(t)dt + \sum_{j=1}^{d} B_{ij}dW_j(t),$$

for some non-diagonal matrices A and B. However, in our models we assume that the matrices A and B are diagonal and studying this more general model would be out of the scope of this thesis.

In contrast to the model in the previous chapter, the background velocity field $(v_1(\mathbf{X}(t), t), v_2(\mathbf{X}(t), t))$ is non-zero and in the higher dimensional cases we study how it affects the Monte Carlo results. In Section 7.1.1 we define the background velocity field as described in Fung and Vassilicos (1998) and Thomson and Devenish (2005).

Finally, we define our initial conditions. In the one-dimensional case we used deterministic initial conditions but for higher dimensional models we study random initial conditions. The reason why we choose random initial conditions is due to the fact that the background velocity is non-zero and we assume that the initial velocity adapts to the ambient velocity of the flow when the particles are released. In particular, we use $\mathbf{X}(0) = (0, 0)$ and $\mathbf{U}(0) = (U_1(0), U_2(0))$ with $U_1(0), U_2(0)$ independent and distributed as $Normal(0, \sigma_U^2)$.

7.1.1 The background velocity field

Our background velocity field $\boldsymbol{v}(\mathbf{X}, t) = (v_1(\mathbf{X}, t), v_2(\mathbf{X}, t))$ describes the wind field as a sum of modes, each with a characteristic length scale L_i and associated wavenumber $k_i = 2\pi/L_i$. The modes have length scales between Λ (the grid size of the weather forecast model) and L (the largest scale, which is of order of the radius of the Earth) and the kinetic energy carried by each of them should be consistent with the energy carried in the real atmosphere. From Nastrom and Gage (1985) the energy spectrum in the atmosphere is given by

$$E(k) = E_0 L(kL)^{-p} \text{ with } 1 \leq p \leq 3, \tag{7.3}$$

for some constant E_0, which means that the energy carried by modes in the range $[k_a, \ldots, k_b]$ is given by

$$\int_{k_a}^{k_b} E(k)dk. \tag{7.4}$$

In addition, the background velocity field should be divergence free.

These lead to the following function from Fung and Vassilicos (1998) and Thomson and Devenish (2005)

$$\boldsymbol{v}(\mathbf{X}, t) = \sum_{n=1}^{N} \left[\mathbf{A}_n \cos(\mathbf{k}_n \cdot \mathbf{X} + \omega_n t) + \mathbf{B}_n \sin(\mathbf{k}_n \cdot \mathbf{X} + \omega_n t)\right], \tag{7.5}$$

which can also be written in complex form as

$$\boldsymbol{v}(\mathbf{X}, t) = \sum_{n=1}^{N} \left[\mathbf{A}_n \operatorname{Re}(g_n) + \mathbf{B}_n \operatorname{Im}(g_n)\right], \tag{7.6}$$

where $g_n = \cos(\mathbf{k}_n \cdot \mathbf{X} + \omega_n t) + i \sin(\mathbf{k}_n \cdot \mathbf{X} + \omega_n t)$. Here, N is the number of modes and for each n we compute a random unit vector $\hat{\mathbf{n}}$ and set $\mathbf{k}_n = k_n \hat{\mathbf{n}}$ where

$$k_n = k_L a^{n-1} \text{ with } a = \left(\frac{L}{\Lambda}\right)^{\frac{1}{N-1}}. \tag{7.7}$$

The vector $\mathbf{k}_n$ is the wavenumber vector of the n^{th} mode, $k_L = \frac{2\pi}{L}$, L is the size of the largest eddy and Λ is the size of the smallest eddy which can be represented by the numerical weather forecast model, i.e. the grid resolution.

The vectors $\mathbf{A}_n$ and $\mathbf{B}_n$ are the mode amplitudes and since we require that the background velocity is incompressible, i.e. $\nabla \cdot \boldsymbol{v} = 0$, it is sufficient to have $\mathbf{k}_n \cdot \mathbf{A}_n = \mathbf{k}_n \cdot \mathbf{B}_n = 0$. Therefore, $\mathbf{A}_n$ and $\mathbf{B}_n$ are chosen to be random but perpendicular to $\mathbf{k}_n$. Their length is given by

$$|\mathbf{A}_n|^2 = |\mathbf{B}_n|^2 = 2E(k_n)\Delta k_n, \tag{7.8}$$

where $E(k)$ is a kinetic energy spectrum and Δk_n is defined by

$$\Delta k_n = \begin{cases} \frac{k_2 - k_1}{2} & \text{for} \quad n = 1, \\ \frac{k_{n+1} - k_{n-1}}{2} & \text{for} \quad 2 \leq n \leq N-1, \\ \frac{k_N - k_{N-1}}{2} & \text{for} \quad n = N. \end{cases} \tag{7.9}$$

Lastly, as in Fung and Vassilicos (1998) we set ω_n (the frequency of the n^{th} mode) to be proportional to the inverse eddy turnover time and use

$$\omega_n = C\sqrt{k_n^3 E(k_n)}, \tag{7.10}$$

where C is a dimensionless constant of proportionality. Equation (7.10) was used for experiments in Fung and Vassilicos (1998) and Thomson and Devenish (2005). In Fung and Vassilicos (1998) they also study the case when $\omega_n = Uk_n$ with a constant velocity U and in Thomson and Devenish (2005) they also consider a stationary field with $\omega_n = 0$. In this thesis we only consider equation (7.10).

7.1.2 σ_U and τ

We now define σ_U and τ which are the last two components to complete our model. The constant σ_U is chosen such that the total kinetic energy carried by the turbulent modes (which are represented by the random part of the SDE) is equal to the kinetic energy of the unresolved modes whose wavenumber vector modulus is greater than $\frac{2\pi}{\Lambda}$. It is then given by

$$\sigma_U^2 = \int_{\frac{2\pi}{\Lambda}}^{\infty} E(k)dk, \tag{7.11}$$

where $E(k)$ is the kinetic energy spectrum given by equation (7.3). This integral evaluates to

$$\sigma_U^2 = E_0 \int_{2\pi L/\Lambda}^{\infty} (kL)^{-p} d(kL) = E_0 \frac{z^{1-p}}{p-1}\Big|_{2\pi L/\Lambda}^{\infty} = \frac{(2\pi)^{1-p}}{p-1} E_0 \left(\frac{\Lambda}{L}\right)^{p-1}. \tag{7.12}$$

The constant τ is chosen to be proportional to the eddy turnover time at the N^{th} mode $k_N = \frac{2\pi}{\Lambda}$ and is given by

$$\tau = \frac{1}{C\sqrt{k_N^3 E(k_N)}}. \tag{7.13}$$

7.1.3 Model scales

Before presenting any numerical results we first analyse the scales of the model variables as we did in Section 2.1.2 for the one-dimensional model. For a physical quantity α we consider a reference scale α^{ref} and write

$$\alpha = \alpha^{\text{ref}}\hat{\alpha}, \tag{7.14}$$

where $\hat{\alpha}$ is the corresponding dimensionless parameter. In particular, we consider a reference length L^{ref}, a reference time T^{ref} and a reference speed U^{ref}. These three quantities are related by (see Section 2.1.2)

$$L^{\text{ref}} = U^{\text{ref}}T^{\text{ref}}. \tag{7.15}$$

For the physical values $\mathbf{X}, \mathbf{U}, L, \Lambda$ and t we then have

$$\mathbf{X} = L^{\text{ref}}\hat{\mathbf{X}}, \quad \mathbf{U} = U^{\text{ref}}\hat{\mathbf{U}}, \quad L = L^{\text{ref}}\hat{L}, \quad \Lambda = L^{\text{ref}}\hat{\Lambda}, \quad t = T^{\text{ref}}\hat{t}, \tag{7.16}$$

where $\hat{\mathbf{X}}, \hat{\mathbf{U}}, \hat{L}, \hat{\Lambda}$ and $\hat{t}$ are the dimensionless parameters.

We set $L^{\text{ref}} = 1km = 1000m$ and we determine U^{ref} by considering E_0 in equation (7.3). Since the energy $E(k)dk$ of a unit particle has the dimension of a squared velocity we have that E_0 has the same dimension which implies that

$$E_0 = (U^{\text{ref}})^2\hat{E}_0. \tag{7.17}$$

Using this equation, we fix U^{ref} such that $\hat{E}_0 = 1$ which gives

$$E_0 = (U^{\text{ref}})^2. \tag{7.18}$$

If we now find the physical value of E_0 we can then compute U^{ref}. In order to find this value we use Figure 4 from Nastrom and Gage (1985) where we obtain $E(10^{-5}m^{-1}) \approx 10^6 m^3 s^{-2}$. Later, in our numerical results we use $L = 1000km = 10^6 m$ and $p = \frac{5}{3}$ so using equation (7.3) gives

$$E_0 \approx 10^{\frac{5}{3}} m^2 s^{-2}. \tag{7.19}$$

Substituting in equation (7.18) we obtain the reference speed

$$U^{\text{ref}} \approx 6.8m/s, \tag{7.20}$$

which implies

$$T^{\text{ref}} \approx 147s. \tag{7.21}$$

As in the one-dimensional case, we drop the hats from these variables and use dimensionless quantities in the equations.

The values of the parameters Λ, number of modes N and travel time T will depend on the experiment since we would like to investigate how the results are affected by the choice of the scale Λ which separates the resolved and unresolved modes. In this thesis, we first fix the largest value of N with its corresponding Λ and then we vary N. In order to keep the modes in the lowest part of the spectrum unchanged we apply the following technique. For a new value of modes $N' < N$ we compute the corresponding Λ' such that

$$k_{N'} = \frac{2\pi}{\Lambda'} = \frac{2\pi}{\Lambda}\left(\frac{L}{\Lambda}\right)^{\frac{N'-1}{N-1}}. \tag{7.22}$$

With this method we guarantee that the first N' modes of the first experiment that uses N and Λ are the same as the modes of the second experiment that uses N' and Λ'. Also, the first N' values of $\mathbf{k}_n$, $\mathbf{A}_n$ and $\mathbf{B}_n$ from the first experiment are the same as the corresponding values from the second experiment. For all random variables involved we are using the same realisations.

7.1.4 Background velocity and spread plots

After defining all components of our two-dimensional model and the model scales we can now construct background velocity and spread plots. A background velocity snapshot can be seen in Figure 7-1 where the arrows indicate the direction and relative magnitude of the background velocity at each point of the domain. For this plot we used the dimensionless parameters $N = 250$, $L = 1000$, $\Lambda = 0.2$, $E_0 = 1$, $p = 5/3$, $C = 1$ and $T = 1$. With the reference scales defined above we have that $L = 1000$ means $1000km$, $\Lambda = 0.2$ means $200m$ and $T = 1$ means $147s$. Note that while our choice of parameters might not represent exactly how the background velocity is observed in practice, our conclusions will not be affected much by choosing a different

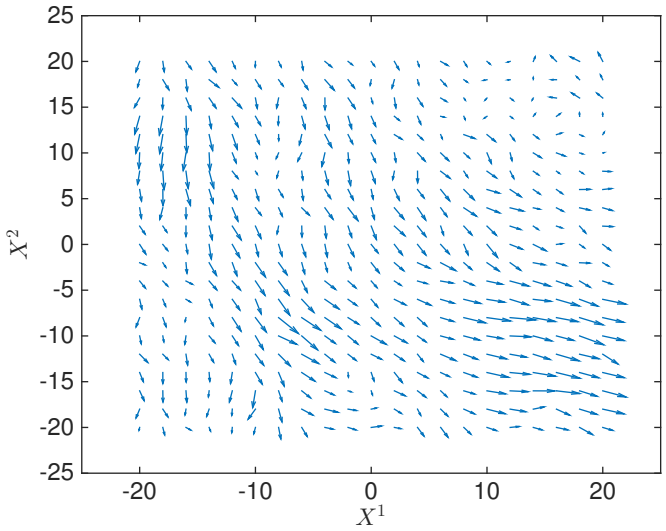


Figure 7-1: A background velocity field snapshot for two-dimensional turbulence.

set of parameters.

Next, we study the spread of the plume with respect to the travel time T. With a non-zero background velocity field we expect the particles to be spread by two processes: (a) turbulence described by the stochastic part of the SDEs and (b) particles getting caught up in different eddies of the background velocity field. Since the particles are initially close together, we expect the effect of (a) to dominate for small times and the effect of (b) to dominate for large times. Our aim for the rest of this section is to determine at what times these different processes dominate, in order to study the performance of MLMC in those very different regimes.

Mathematically the spread is described by the standard deviation of the particle's position, which is estimated using

$$s = \sqrt{\frac{1}{N_L - 1} \sum_{j=1}^{N_L} \left[\left(X_{1\,M_L}^{(j)} - \mu_1 \right)^2 + \left(X_{2\,M_L}^{(j)} - \mu_2 \right)^2 \right]}, \tag{7.23}$$

where

$$\mu_i = \frac{1}{N_L} \sum_{j=1}^{N_L} X_{i\,M_L}^{(j)}, \quad i = 1, 2, \tag{7.24}$$

is an estimate for the mean particle position. The value $X_{i\,M_L}^{(j)}$ is an approximation of the j^{th} independent sample path of $X_i(T)$ using some timestepping method with M_L

timesteps.

For the one-dimensional homogeneous model with zero background velocity studied in Section 2.1.1 (equations (2.13) and (2.8)) it is possible to derive an exact expression for the spread s. Details of the derivation can be found in Appendix A.2.2 from which we obtain

$$s \sim \begin{cases} t & \text{for} \quad t \ll \tau, \\ \sqrt{t} & \text{for} \quad t \gg \tau, \end{cases}$$

for stochastic $U(0) \sim Normal(0, \sigma_U^2)$ and

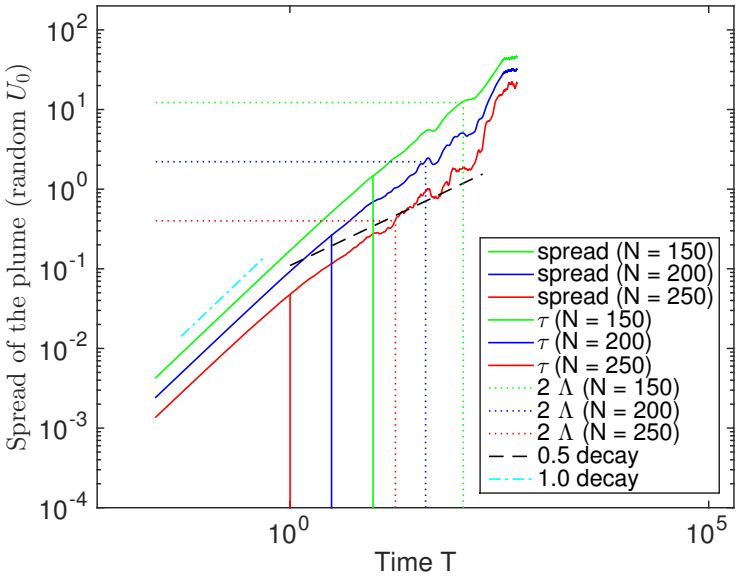
$$s \sim \begin{cases} t^{\frac{3}{2}} & \text{for} \quad t \ll \tau, \\ \sqrt{t} & \text{for} \quad t \gg \tau, \end{cases}$$

for constant $U(0)$. The same behaviour can also be obtained theoretically for a two-dimensional homogeneous model with zero background velocity (i.e. $v = 0$ in equations (7.1) and (7.2)). The result for stochastic $U(0)$ agrees with the analysis from Taylor (1921) and the short time behaviour for constant $U(0)$ agrees with Richardson's t^3 law for s^2.

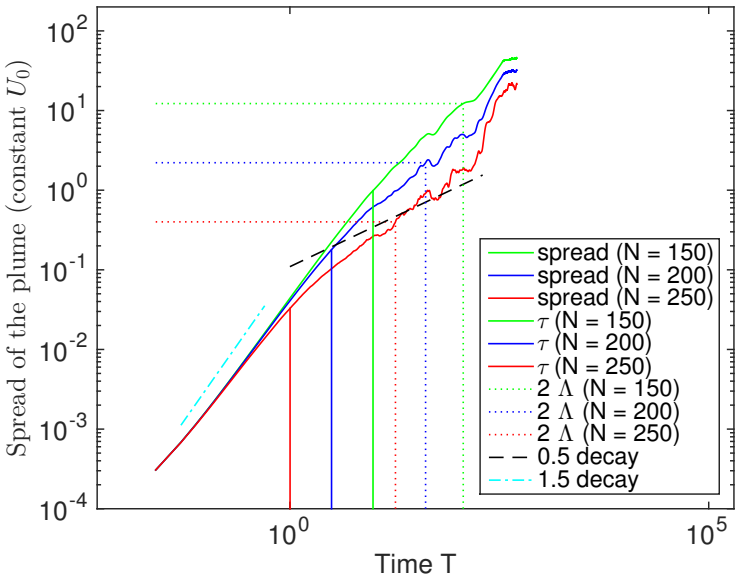
For the two-dimensional model described by equations (7.1) and (7.2) with non-zero background velocity it is difficult to obtain a theoretical expression for the spread so we just approximate it numerically. However, for small travel times we expect a similar behaviour as in the case of a zero background velocity (since the particles have not been separated in different eddies yet) and for large travel times we expect a further increase in the particle spread.

In Figure 7-2 we plot the spread with respect to time T for random and constant $\mathbf{U}(0)$ using the Symplectic Euler method. The plots contain data for three different values of N (and also different Λ as discussed at the end of Section 7.1.3) and the solid vertical lines indicate the value of τ that corresponds to each value of N. The horizontal dotted lines represent the value of 2Λ since we are interested to see at what times the spread becomes larger than 2Λ where Λ is the radius of the smallest eddy corresponding to each number of modes N.

For random $\mathbf{U}(0)$ the rate is initially linear and for constant $\mathbf{U}(0)$ it is initially proportional to $t^{\frac{3}{2}}$. Therefore, the short time behaviour agrees with the theoretical results of the simpler problem considered in Appendix A.2.2. After approximately time τ (which depends on Λ as can be seen from equation (7.13)) the behaviour for



(a) Random $\mathbf{U}(0)$



(b) Constant $\mathbf{U}(0)$

Figure 7-2: Spread plots with respect to time for equations (7.1) and (7.2).

both initial conditions is similar. We observe that for $N = 250$ and $N = 200$ the rate becomes proportional to the square root of t.

For very large values of T the rate becomes linear again and it is also noisy as it can be seen from the plot. The reason that it becomes linear is due to the fact that for large times T the particles are separated in different eddies of the background velocity field which causes an increase in the rate. This separation is expected to happen at times when the spread exceeds the value 2Λ (since Λ is the radius of the smallest eddy) which is also the case (approximately) for our model as it can be seen from the plot. For $N = 150$ the rate only becomes linear after time τ.

7.1.5 Numerical results

In this section we present some Monte Carlo results in a similar way as in Chapter 6 for the one-dimensional model. We approximate the solution of the two-dimensional model given by equations (7.1) and (7.2) with τ, σ_U and $\boldsymbol{v}(\mathbf{X}, t)$ as in equations (7.13), (7.11) and (7.5) respectively. The model parameters are $N = 250$, $L = 1000$, $\Lambda = 0.2$, $E_0 = 1$, $p = 5/3$ and $C = 1$. For initial conditions we use $\mathbf{X}(0) = (0, 0)$ and $\mathbf{U}(0) = (U_1(0), U_2(0))$ with $U_1(0), U_2(0)$ independent and distributed as $Normal(0, \sigma_U^2)$.

Since our model is homogeneous the profiles of σ_U and τ (see Section 7.1.2) remain constant and they do not become large as in the one-dimensional inhomogeneous model considered in Chapter 6. As a result, we do not expect any large improvements in the Monte Carlo performance when using Geometric Langevin or BAOAB methods. Therefore, we only consider the Symplectic Euler method which is currently used by the Met Office and concentrate on investigating the gains of MLMC compared to StMC.

In order to choose suitable travel times we use the spread plot given in Figure 7-2a. We consider two travel times, $T_1 \approx \frac{\tau + T_\Lambda}{2}$ and $T_2 \approx \frac{3}{2} T_\Lambda$ where T_Λ is defined as the time when the spread becomes larger than 2Λ. We choose the first time T_1 in order to compare results when the spread is linear and the particles are not yet separated into different eddies which means that we are in the regime where the spread is due to turbulence. The second time $T_2 > T_\Lambda$ is chosen in order to study how the Monte Carlo methods are affected when the particles are separated in different eddies (since Λ is the radius of the smallest eddy). In particular, when the travel time is larger than T_Λ it might be harder to couple the random variables in the MLMC coarse step since

the fine and coarse trajectories might be separated into different eddies.

For our given set of parameters we have that $\tau \approx 1.00$ and from the plot $T_\Lambda \approx 18.2$ and therefore we use $T_1 = 10$ and $T_2 = 28$. From the model scales Section 7.1.3 and since $T^{\text{ref}} \approx 147s$ we have that T_1 corresponds to $1470s = 24.5min$ of travel time and T_2 corresponds to $4116s = 68.6min$ of travel time.

Our quantity of interest is the particle concentration in a two-dimensional box $[a_1, b_1] \times [a_2, b_2]$ at the final time T. In mathematical terms we would like to approximate $\mathbb{E}[\mathbb{1}_{[a_1,b_1]\times[a_2,b_2]}(\mathbf{X}(T))]$ where

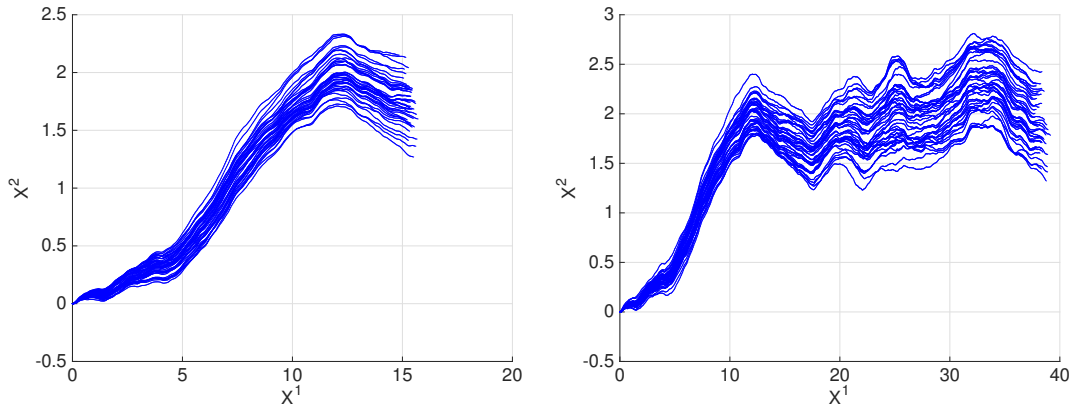
$$\mathbb{1}_{[a_1,b_1]\times[a_2,b_2]}(\mathbf{x}) = \begin{cases} 1 & \text{if} \quad \mathbf{x} \in [a_1, b_1] \times [a_2, b_2], \\ 0 & \text{otherwise}, \end{cases} \tag{7.25}$$

is the two-dimensional indicator function. However, the discontinuity of the indicator function causes an increase in the MLMC variance and therefore we would like to use a smooth approximation as discussed in Section 5.4 for the one-dimensional case. In order to construct this smooth approximation we first write equation (7.25) in the following form

$$\mathbb{1}_{[a_1,b_1]\times[a_2,b_2]}(\mathbf{x}) = \mathbb{1}_{[a_1,b_1]}(x_1) \cdot \mathbb{1}_{[a_2,b_2]}(x_2) = \begin{cases} 1 & \text{if} \quad \mathbf{x} \in [a_1, b_1] \times [a_2, b_2], \\ 0 & \text{otherwise}, \end{cases} \tag{7.26}$$

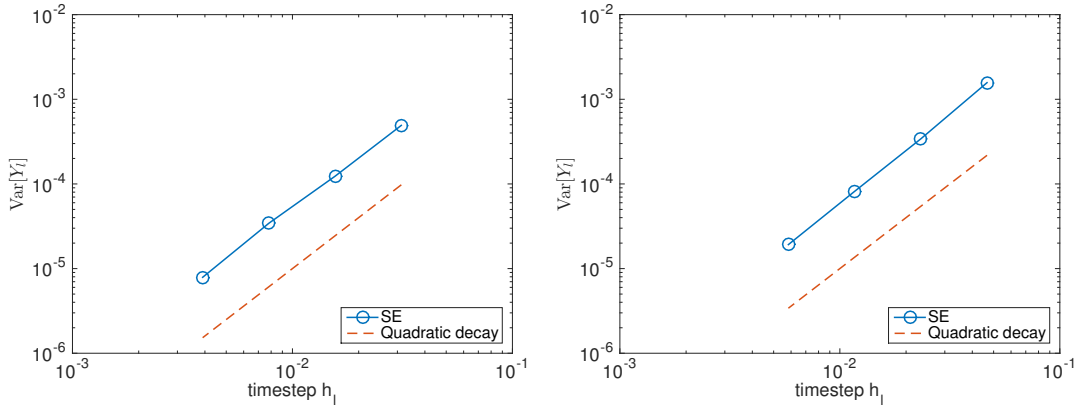
with $\mathbf{x} = (x_1, x_2)$. Then, we approximate each of the functions $\mathbb{1}_{[a_1,b_1]}(x_1)$ and $\mathbb{1}_{[a_2,b_2]}(x_2)$ by a smoothing polynomial (Giles et al., 2015) in exactly the same way as we did in Section 5.4.1. Our quantity of interest then simply becomes the product of two smoothed indicator functions. We also use the same smoothing polynomial parameters δ and r, i.e. $\delta = 0.1$ and $r = 4$. For $T_1 = 10$ the two-dimensional box is $[15.26, 15.46] \times [1.68, 1.78]$ and for $T_2 = 28$ is $[37.85, 38.85] \times [1.87, 1.97]$. These boxes are chosen such that they are around the mean particle position and contain approximately 10% of the released particles.

Before we present our Monte Carlo results we first construct some trajectory plots for our two travel times $T_1 = 10$ and $T_2 = 28$. In Figure 7-3 we construct two trajectory plots using the Symplectic Euler method and each plot contains 40 trajectories. We observe that in both cases the particles initially form a narrow plume and at later times they start spreading.



(a) $T_1 = 10$ (b) $T_2 = 28$

Figure 7-3: Two-dimensional trajectory plots

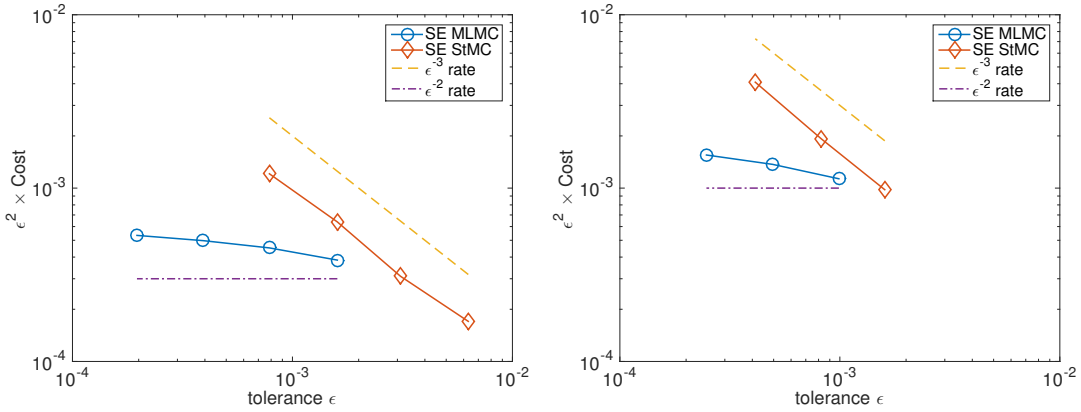


(a) $T_1 = 10$ (b) $T_2 = 28$

Figure 7-4: Variance rates - Concentration

Finally, we present our Monte Carlo results. We compare Standard and Multilevel Monte Carlo with the Symplectic Euler (SE) timestepping method. In Figure 7-4 we firstly plot $Var[Y_l]$ as a function of the timestep size h_l. We observe that for both travel times the variance rate is proportional to h_l^2 which confirms condition (iii) of Theorem 3.1 (complexity theorem). Therefore we expect to see $\mathcal{O}(\epsilon^{-2})$ cost rates for MLMC.

In Figure 7-5 we plot the total computational cost of StMC and MLMC. The cost is plotted as a function of the root mean square error tolerance ϵ. The MLMC coarsest level timestep size for $T_1 = 10$ is $h_0 = \frac{T_1}{180} \approx 5.56 \cdot 10^{-2}$ and for $T_2 = 28$ is $h_0 = \frac{T_2}{400} \approx 7.0 \cdot 10^{-2}$. For the two computational times we observe a similar behaviour



(a) $T_1 = 10$ (b) $T_2 = 28$

Figure 7-5: Cost rates - Concentration

but for $T_2 = 28$ the overall cost is higher. The MLMC cost rates are asymptotically proportional to ϵ^{-2} as expected from the quadratic variance decay rates shown in Figure 7-4. The StMC cost rates are asymptotically proportional to ϵ^{-3} which follows from the discussion in Section 3.2.1 and equation (3.20).

For the travel time $T_1 = 10$ we can also see that when the tolerance is approximately larger than $3 \cdot 10^{-3}$ is better to use the StMC but for smaller tolerances the MLMC method becomes faster. For $T_2 = 28$ the MLMC method becomes faster when the tolerance is approximately smaller than $1 \cdot 10^{-3}$. Now if we compare the two travel times, we can see that for $T_1 = 10$ the MLMC method gives a benefit for a larger tolerance and this is probably due to the fact that there is less separation of particles in different eddies.

Finally, note that the MLMC cost also depends on the timestep size of the coarsest level. When Theorem 3.1 holds the cost is concentrated on the coarsest level and therefore by increasing the corresponding timestep size (and also the number of levels) we can reduce the overall cost. However, we might need a smaller tolerance to obtain the ϵ^{-2} asymptotic behaviour of the cost and in addition if the timestep size is too large then the Symplectic Euler method will become unstable (see also Section 3.1 for a discussion on instability). The effect on the cost of the coarsest level timestep size is studied in Section 7.3.

7.2 The three-dimensional model

Next, we study a three-dimensional model. For the horizontal direction we assume that the motion is described by the two-dimensional model of the previous section with the addition of vertical wind-shear and for the vertical direction we assume that the motion is described by the model from Chapter 6. In particular, we assume that σ_U and τ vary strongly with height. In this section we denote by $\mathbf{U}(t) = (U_1(t), U_2(t), U_3(t))$ and $\mathbf{X}(t) = (X_1(t), X_2(t), X_3(t))$ the three-dimensional velocity and position respectively. The first two components denote the horizontal motion and third component denotes the vertical motion. The model is given by the following SDEs

$$dU_i(t) = -\frac{U_i(t)}{\tau_{ho}}dt + \sqrt{\frac{2\sigma^2_{U,ho}}{\tau_{ho}}}dW_i(t), \tag{7.27}$$

$$dX_i(t) = (v_i(\mathbf{X}(t), t) + U_i(t))\, dt, \quad i = 1, 2, \tag{7.28}$$

$$dU_3(t) = -\lambda(X_3(t))U_3(t)dt - \frac{\partial V}{\partial X_3}(X_3(t), U_3(t))dt + \sigma(X_3(t))dW_3(t), \tag{7.29}$$

$$dX_3(t) = U_3(t)dt, \tag{7.30}$$

where

$$\lambda(X) = \frac{1}{\tau_{ve}(X)}, \quad \sigma(X) = \sqrt{\frac{2\sigma^2_{U,ve}(X)}{\tau_{ve}(X)}}, \tag{7.31}$$

$$\frac{\partial V}{\partial X}(X, U) = -\frac{1}{2}\left[1 + \left(\frac{U}{\sigma_{U,ve}(X)}\right)^2\right]\frac{\partial \sigma^2_{U,ve}}{\partial X}(X), \tag{7.32}$$

and $\mathbf{W}(t) = (W_1(t), W_2(t), W_3(t))$ is a three-dimensional Brownian motion. The notation "ho" is used to denote the horizontal profiles and the notation "ve" is used to denote the vertical profiles.

Equations (7.27) and (7.28) are as in the two-dimensional model considered in Section 7.1 and we use the same profiles for τ_{ho} and $\sigma_{U,ho}$. In particular we use

$$\sigma^2_{U,ho} = \frac{(2\pi)^{1-p}}{p-1}E_0\left(\frac{\Lambda}{L}\right)^{p-1}, \quad \tau_{ho} = \frac{1}{C\sqrt{k_N^3 E(k_N)}}, \tag{7.33}$$

where $E(k) = E_0 L(kL)^{-p}$ and Λ, L, E_0, p, C and k_N are constants. The background velocity field $(v_1(\mathbf{X}(t), t), v_2(\mathbf{X}(t), t))$ will be slightly different than the one used be-

fore and it is defined in Section 7.2.1. Also, it is applied only on the horizontal plane since in reality the vertical background velocities are much smaller than the horizontal velocities and can be neglected.

For the vertical profiles $\sigma_{U,ve}$ and τ_{ve} we use the same functions as in Section 2.2 which are given by

$$\sigma_{U,ve}(X) = \kappa_\sigma u^* \left(1 - \frac{X}{H}\right)^{\frac{3}{4}}, \quad \tau_{ve}(X) = \kappa_\tau \frac{X}{\sigma_{U,ve}(X)}, \quad X \in (0, H), \tag{7.34}$$

where $\kappa_\sigma, \kappa_\tau, u^*$ and H are constants. The Met Office's regularisation given by equations (2.24) and (2.25) is also applied in this case. Note that for the horizontal profiles we could use similar functions as in (7.34) (Webster et al., 2003) but for simplicity we assume the same profiles as in the two-dimensional case.

Next, we define the boundary conditions. On the horizontal plane we assume that there are no boundary conditions and that the particles can travel anywhere as in Section 7.1. On the vertical motion however, we assume reflective boundary conditions as in the one-dimensional inhomogeneous model. When a particle hits the lower boundary ($X_3 = 0$) or the upper boundary ($X_3 = H$) it is reflected elastically as described by equations (2.26) and (2.27).

Finally, we define the initial conditions. As in the two-dimensional case we use $\mathbf{X}(0) = (0, 0, X_3(0))$ and $\mathbf{U}(0) = (U_1(0), U_2(0), U_3(0))$ with $U_1(0), U_2(0), U_3(0)$ independent and normally distributed. $U_1(0), U_2(0)$ and distributed as $Normal(0, \sigma^2_{U,ho})$ and $U_3(0)$ is distributed as $Normal(0, \sigma^2_{U,ve}(X_3(0)))$. The value of $X_3(0)$ will be constant.

7.2.1 The background velocity field

For the background velocity field of the three-dimensional model we use the same equations as in the two-dimensional case but with some changes in order to add dependence on the height. For this, we apply the analysis presented in Holton (2004). Writing all the details is beyond the scope of this thesis so we just mention some of the key results and assumptions.

The author of Holton (2004) begins from the Navier-Stokes equations and takes into account two forces that act on a parcel of air in the boundary layer, the viscous shear force and the Coriolis force. Then, they make a series of assumptions like for

example assuming a horizontally homogeneous turbulence (as in our case) and making the Boussinesq approximation. With the Boussinesq approximation the density is assumed to be constant everywhere, except for the terms where it appears multiplied by the gravity's acceleration constant (the buoyancy terms). They also assume the no-slip boundary condition at the bottom of the boundary layer, i.e. that the background velocity is zero at that point. Then, they assume the existence of a horizontal background velocity field $(v_{1,g}(X_1, X_2, t), v_{2,g}(X_1, X_2, t))$, called the geostrophic velocity of the free atmosphere, that depends only on the horizontal variables. In our case this field can be the one that we have already used for the two-dimensional model which is given by equation (7.5).

With the above assumptions, they then derive the following equations of what is called the Ekman layer

$$K_m \frac{\partial^2 v_1}{\partial x_3^2}(x_1, x_2, x_3, t) + c(v_2(x_1, x_2, x_3, t) - v_{2,g}(x_1, x_2, t)) = 0, \tag{7.35}$$

$$K_m \frac{\partial^2 v_2}{\partial x_3^2}(x_1, x_2, x_3, t) - c(v_1(x_1, x_2, x_3, t) - v_{1,g}(x_1, x_2, t)) = 0, \tag{7.36}$$

where K_m (eddy viscosity) and c (Coriolis parameter) are constants. The solution $\boldsymbol{v}(\mathbf{X}, t) = (v_1(\mathbf{X}, t), v_2(\mathbf{X}, t))$ of equations (7.35) and (7.36) will then be the background velocity field that we use in the model equation (7.28).

With the no-slip assumption at the bottom of the boundary layer the following holds

$$v_1(x_1, x_2, x_3, t) = v_2(x_1, x_2, x_3, t) = 0 \quad \text{when} \quad x_3 = 0. \tag{7.37}$$

Also, in Holton (2004) they assume that $(v_1(\mathbf{X}, t), v_2(\mathbf{X}, t))$ approaches $(v_{1,g}, v_{2,g})$ when you are far from the ground which gives the following conditions

$$\begin{aligned} v_1(x_1, x_2, x_3, t) &\to v_{1,g}(x_1, x_2, t), \quad \text{as} \quad x_3 \to \infty \\ v_2(x_1, x_2, x_3, t) &\to v_{2,g}(x_1, x_2, t), \quad \text{as} \quad x_3 \to \infty \end{aligned} \tag{7.38}$$

The system of ODEs (7.35) and (7.36) with the boundary conditions (7.37) and (7.38) can now be solved (Holton, 2004) to give

$$v_1 = v_{1,g}(1 - e^{-\gamma x_3}\cos(\gamma x_3)) - v_{2,g}e^{-\gamma x_3}\sin(\gamma x_3), \tag{7.39}$$

$$v_2 = v_{2,g}(1 - e^{-\gamma x_3}\cos(\gamma x_3)) - v_{1,g}e^{-\gamma x_3}\sin(\gamma x_3), \tag{7.40}$$

where $\gamma = \sqrt{\frac{c}{2K_m}}$. Equations (7.39) and (7.40) are called the Ekman spiral and they lead to wind-shear, i.e. a variation of the horizontal velocity with height. The physical value of K_m is $5m^2s^{-1}$, c is given by $10^{-4}s^{-1}$ and γ by $3 \cdot 10^{-3}m^{-1}$. With $(v_{1,g}, v_{2,g})$ as in (7.5) we now have our background velocity field.

The model scales for the three-dimensional model are the same as the ones used for the two-dimensional model in the previous section.

7.2.2 Numerical results

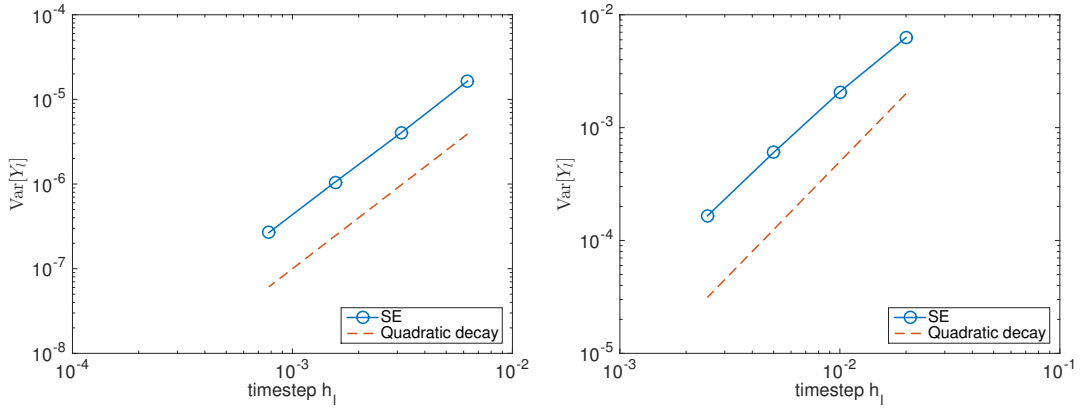
Finally, we present some numerical results. As in the two-dimensional case we consider the Standard and Multilevel Monte Carlo methods with Symplectic Euler (SE). Our model parameters are $N = 250$, $L = 1000$, $\Lambda = 0.2$, $E_0 = 1$, $p = 5/3$ and $C = 1$ for the horizontal motion and $\kappa_\sigma = 1.3, \kappa_\tau = 0.5, u_* = 0.2, H = 1$ and $\epsilon_{\text{reg}} = 0.01$ for the vertical motion. These are the same parameters used for the numerical results in Sections 7.1 and 6.1 respectively. For the background velocity field equations (7.39) and (7.40) we use $\gamma = 3$. Our initial conditions are $\mathbf{X}(0) = (0, 0, 0.05)$ and $\mathbf{U}(0) = (U_1(0), U_2(0), U_3(0))$ with $U_1(0), U_2(0), U_3(0)$ independent and normally distributed. $U_1(0), U_2(0)$ and distributed as $Normal(0, \sigma^2_{U,ho})$ and $U_3(0)$ is distributed as $Normal(0, \sigma^2_{U,ve}(0.05))$. Our travel times are $T_1 = 1$ and $T_2 = 8$ which correspond to approximately $147s = 2.45min$ and $8 \cdot 147s = 19.6min$ respectively.

Our quantity of interest is the particle concentration in a three-dimensional box $[a_1, b_1] \times [a_2, b_2] \times [a_3, b_3]$ which mathematically means that we approximate $\mathbb{E}[\mathbb{1}_{[a_1,b_1]\times[a_2,b_2]\times[a_3,b_3]}(\mathbf{X}(T))]$ where

$$\mathbb{1}_{[a_1,b_1]\times[a_2,b_2]\times[a_3,b_3]}(\mathbf{x}) = \begin{cases} 1 & \text{if} \quad \mathbf{x} \in [a_1, b_1] \times [a_2, b_2] \times [a_3, b_3], \\ 0 & \text{otherwise.} \end{cases} \tag{7.41}$$

As before, we approximate the three dimensional indicator function (7.41) by smoothing polynomials with parameters $\delta = 0.1$ and $r = 4$. For $T_1 = 1$ the three-dimensional box is $[0.15, 0.25] \times [0.15, 0.25] \times [0.04, 0.07]$ and for $T_2 = 8$ is $[2.6, 3.2] \times [2.1, 2.7] \times [0.06, 0.22]$. These boxes are chosen with the same approach as before, i.e. they are around the mean particle position and contain approximately 10% of the released particles.

In Figure 7-6 we firstly plot the variance $Var[Y_l]$ as a function of the timestep size h_l. We observe that in both cases the variance decays with a quadratic rate which



(a) $T_1 = 1$ (b) $T_2 = 8$

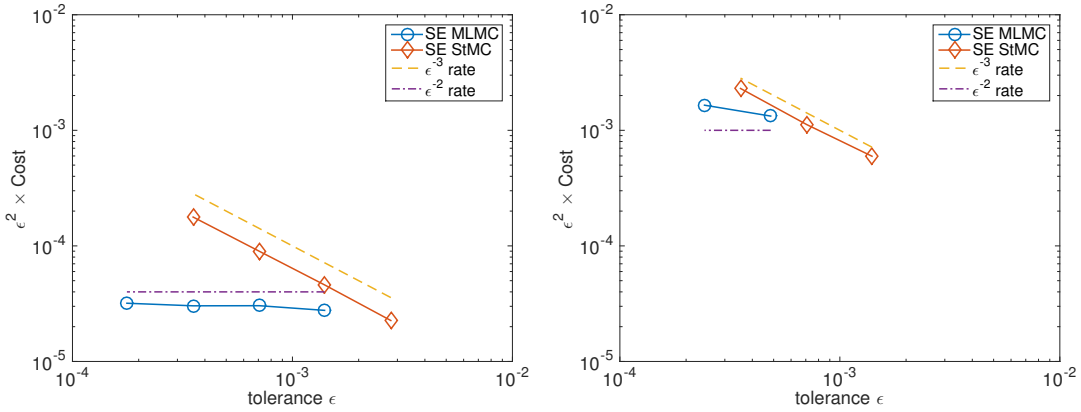
Figure 7-6: Variance rates - Concentration

confirms condition (iii) of Theorem 3.1. Therefore we expect to see $\mathcal{O}(\epsilon^{-2})$ cost rates for MLMC.

In Figure 7-7 we plot the total computational cost of StMC and MLMC as a function of the root mean square error tolerance ϵ. The MLMC coarsest level timestep size for $T_1 = 1$ is $h_0 = \frac{T_1}{40} = 2.5 \cdot 10^{-2}$ and for $T_2 = 8$ is $h_0 = \frac{T_2}{700} \approx 1.14 \cdot 10^{-2}$. For both travel times the StMC cost rates are asymptotically proportional to ϵ^{-3} as expected from the discussion in Section 3.2.1. Also for $T_1 = 1$ the MLMC cost rates are asymptotically proportional to ϵ^{-2} which follows from Theorem 3.1 and the quadratic variance rates shown in Figure 7-6a.

However for $T_2 = 8$ the MLMC cost rates are not asymptotically proportional to ϵ^{-2}. One of the reasons for this is that the timestep size h_0 of the coarsest level is not small enough. By decreasing h_0 the cost rates become quadratic but at the same time the overall cost increases. The effect that h_0 has on the MLMC performance will be examined in Section 7.3. Also, by adding more data points for smaller ϵ in Figure 7-7b or by increasing the number of MLMC levels would possibly help in the improvement of the cost rate, but both approaches will require runs with large computational times.

Our conclusion for $T_1 = 1$ is that when ϵ is approximately larger than $2 \cdot 10^{-3}$ is better is use the StMC but for smaller tolerances the MLMC method is faster. For $T_2 = 8$ the MLMC method becomes faster when the tolerance is approximately smaller than $5 \cdot 10^{-4}$. As we will see in Section 7.3.1, increasing h_0 reduces the orerall MLMC cost but at the same time it has a small negative impact on the cost rates. As a result, MLMC can be faster for larger values of ϵ since the two lines representing the StMC



(a) $T_1 = 1$ (b) $T_2 = 8$

Figure 7-7: Cost rates - Concentration

and MLMC cost will intercept earlier.

7.3 Studying the effect of changing some model parameters on the Monte Carlo performance

7.3.1 Varying the coarsest level timestep size

In this section we study how the MLMC performance is affected when we vary the timestep size on the coarsest level. Since the MLMC cost is concentrated on the coarsest level we expect to see a lower cost when we increase the size of the corresponding timestep. The Symplectic Euler method however can be unstable when the timestep size on every level is not small enough, so we also show results for BAOAB which does not have any stability constraints. For all numerical experiments we use the three-dimensional model and exactly the same model parameters as those used in Section 7.2. We only vary the number of timesteps M_0 on the coarsest level and we consider the travel time $T = 1$. The timestep on the coarsest level is denoted by $h_0 = \frac{T}{M_0}$.

In Figure 7-8 we plot the same Symplectic Euler cost rates as in Figure 7-7a together with the corresponding BAOAB cost rates. For both timestepping methods we have $M_0 = 40$. We observe that BAOAB-MLMC is more expensive than Symplectic Euler-MLMC and for both methods the cost rates are asymptotically proportional to $\mathcal{O}(\epsilon^{-2})$.

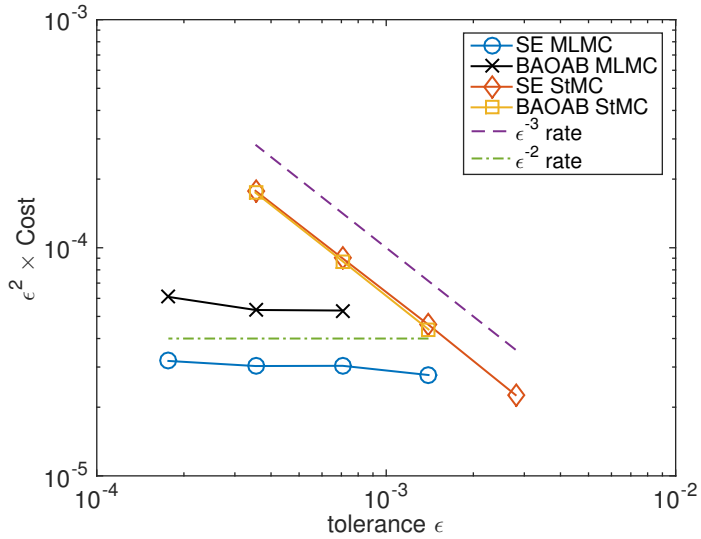
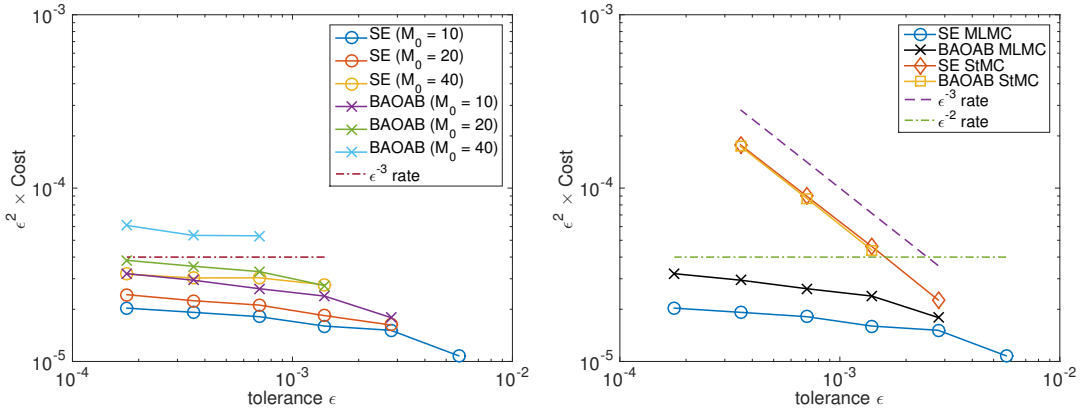


Figure 7-8: Symplectic Euler and BAOAB cost rates - Concentration

Also note that BAOAB-StMC has the same cost as Symplectic Euler-StMC. With this choice of model parameters, the two timestepping methods do not have a large difference in the bias error as in the one-dimensional case and therefore they have a similar cost. In Section 7.3.2 we change some values of the model parameters and study when the BAOAB method can be faster than Symplectic Euler.

In Figure 7-9a we plot MLMC cost rates for different values of M_0. For both timestepping methods, decreasing the size of M_0 (which increases the coarsest timestep h_0) results in a reduction of the overall cost. As it can also be seen from the plot, this change has a negative impact on the rates of the cost which become slightly larger than quadratic. However there is still a lot of benefit, especially for larger ϵ, as can be seen in Figure 7-9b. In Figure 7-9b we plot the same StMC cost values as in Figure 7-8 together with the MLMC cost rates that correspond to $M_0 = 10$. We observe that for all the values of ϵ considered, we have that the MLMC method is faster than StMC. Therefore, it is possible for MLMC to become faster than StMC for larger values of ϵ than those discussed at the end of Section 7.2.

If we compare only the MLMC costs from Figure 7-9b we can see that Symplectic Euler is still faster than BAOAB when M_0 is smaller. However, it is very important to be careful with any further decrease of M_0 since there are stability constaints for the Symplectic Euler timestep size (see Section 3.1). For BAOAB there are no stability constaints so it is possible to achieve further reductions in the cost by decreasing M_0.



(a) MLMC for different M_0

(b) StMC-MLMC cost rates

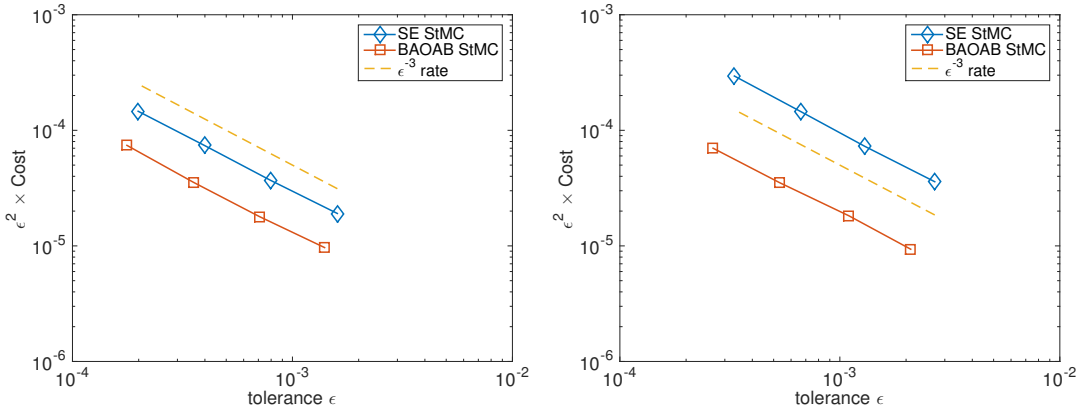
Figure 7-9: Monte Carlo cost rates

7.3.2 Varying the model parameters

In the last section we change some of the model parameters and study a few scenarios where we expect BAOAB-StMC to perform better than Symplectic Euler-StMC. The first change is to reduce the value of the regularisation height to $\epsilon_{\text{reg}} = 0.002$ from $\epsilon_{\text{reg}} = 0.01$ (see equations (2.24) and (2.25)). As discussed at the end of Section 2.2, the function $\lambda(X)$ present in the drift term of the vertical component of our three-dimensional model (equation (7.29)) is bounded above by $1/\tau(\epsilon_{\text{reg}})$. Reducing the value of ϵ_{reg} increases the upper bound $1/\tau(\epsilon_{\text{reg}})$ and therefore the function $\lambda(X)$ can take larger values. This property is then combined with the modified equations analysis from Section 5.3.2 where we show that when $\lambda(X)$ is large the BAOAB method is expected to work better than Symplectic Euler.

Our next change is to reduce the height where the particles are reflected. Instead of reflecting at the height $X_3 = 1$ we reflect at $X_3 = 0.1$ in order for the particles to spend more time at lower heights where $\lambda(X)$ is larger. Also, we reduce the particle release height to $X_3 = 0.02$ from $X_3 = 0.05$ for the same reason. We do not show any results for MLMC since we have already demonstrated that it has an asymptotically better cost rate when compared to StMC.

In Figure 7-10a we plot the StMC cost rates for BAOAB and Symplectic Euler. We observe that in this case the cost of BAOAB is smaller and both methods have an asymptotic rate proportional to ϵ^{-3}. This is the behaviour that we expected since the particles spend more time near the lower boundary where $\lambda(X)$ is large. The cost of



(a) $X_3(0) = 0.02$ and $\epsilon_{\text{reg}} = 0.002$ (b) $X_3(0) = 0.01$ and $\epsilon_{\text{reg}} = 0.001$

Figure 7-10: StMC cost rates with lower regularisation, reflection and release heights

Symplectic Euler is approximately $2\times$ larger than the cost of BAOAB.

In Figure 7-10b we reduce the regularisation and release height even further and we observe an increase in the cost difference between the two timestepping methods. This observation was expected since the function $\lambda(X)$ can now take larger values which slows down the Symplectic Euler method. The cost of Symplectic Euler is now approximately $5\times$ larger than the cost of BAOAB.

The conclusion from the numerical results of Figure 7-10 is that BAOAB-StMC can be very useful in situations where the particles spend a lot of time in regions where $\lambda(X)$ is large. In addition, when the physical problem under consideration does not have this property, we observed that BAOAB-StMC has the same cost as Symplectic Euler-StMC (see Figure 7-8) and therefore using BAOAB can only result in an improvement in the cost.

Our overall conclusion is similar to the one-dimensional case. For large tolerances we recommend using BAOAB-StMC and for small tolerances we recommend using Symplectic Euler-MLMC.

Chapter 8

Conclusion

In this thesis we presented several improvements to the StMC-Symplectic Euler algorithm currently used by the Met Office to problems from atmospheric dispersion modelling. We firstly considered a one-dimensional model for vertical dispersion on a fixed boundary layer and computed the particles' mean position, concentration and density function. We demonstrated that with improved timestepping methods and in particular with Geometric Langevin and BAOAB, we can achieve significant speed-ups for the StMC method when compared to using Symplectic Euler. In particular, Geometric Langevin was approximately $13\times$ faster and BAOAB was approximately $52\times$ faster. The main reason for this improvement is due to the fact that Geometric Langevin and BAOAB can work better when the term $\lambda(X)$ (present in the drift term of our model) becomes large which reduces the bias error.

In order to produce results for the particles' concentration and density function we used polynomial approximations to the indicator function as described in Giles et al. (2015). For these approximations it was necessary to choose a priori two values for some parameters δ and r and we then confirmed numerically that for our choice of parameters the error introduced by the polynomial approximation is negligible when compared to the bias error. The parameters δ and r can also be computed by using on-the-fly estimators and the reader is referred to Giles et al. (2017) for more details. This application can be a topic for future research.

We then developed and implemented a new algorithm for the treatment of reflective boundary conditions which preserves the quadratic variance decay of the MLMC method. With this new treatment, which simply changes the sign of the random vari-

ables when is necessary, we were able to show numerically that the MLMC method improves the asymptotic cost rates. In particular for a given tolerance ϵ the corresponding cost rate of the StMC method is asymptotically proportional to ϵ^{-3} and MLMC method improves this rate to ϵ^{-2}. The MLMC method can therefore be very useful especially for small tolerances. However, for large tolerances we recommend the use of StMC with BAOAB since it can be faster than MLMC.

All the above results were also proved theoretically with the addition of suitable assumptions. We firstly proved existence and uniqueness of a solution to the one-dimensional model by using a Lyapunov function and the theory from Khasminskii (2011). Then, by using the theory from Milstein and Tretyakov (2005) and the same Lyapunov function, we proved the convergence of our timestepping methods. The key idea for this proof was to reject any trajectories that go outside a fixed domain.

Our next theoretical result was to prove the application of Threorem 3.1 (complexity theorem (Giles, 2008)) to the one-dimensional models and in particular proving that MLMC improves the asymptotic cost rates. We applied a direct approach based on strong approximation results for a simplified model and a modified equations (Shardlow, 2006), (Zygalakis, 2011), (Müller et al., 2015) approach for our inhomogeneous model. In the case of the inhomogeneous model we computed modified equations for every timestepping method, which also showed that BAOAB is almost a second order method and therefore explain why it has a smaller bias error.

Our final result for the one-dimensional models was the implementation of an adaptive MLMC algorithm. For Symplectic Euler this gave some small improvements but for Geometric Langevin we observed some counter-intuitive results such as an increase in the bias error when reducing the average size of the timesteps. A further investigation to study these unexpected results can be a topic for future research.

Next, we considered higher dimensional models. Firstly, we studied a two-dimensional model for homogeneous turbulence on the horizontal plane. We implemented MLMC and showed numerically that improves the asymptotic cost rate to ϵ^{-2} from ϵ^{-3} which is the corresponding rate of StMC. Then, we studied a three-dimensional model which is a combination of the horizontal structure of the two-dimensional model with the vertical structure of the one-dimensional inhomogeneous model studied earlier. We observed that in this case MLMC also improves the asymptotic cost rate to ϵ^{-2} and in addition we demonstrated that in some special cases we have that the BAOAB method reduces the overall cost of StMC.

As mentioned above, one of the basic outcomes of this thesis is to show that MLMC can reduce the asymptotic cost rate from ϵ^{-3} to ϵ^{-2} which is the best that can be achieved for a Monte Carlo method. A possible direction for future research can therefore be the study of Quasi-Monte Carlo methods which can reduce this rate even further. In particular, with the Multilevel Quasi-Monte Carlo (MLQMC) method we can reduce this rate to $\epsilon^{-\alpha}$ for some $1 < \alpha < 2$ that depends on the problem. Examples of Standard Quasi-Monte Carlo and MLQMC can be found in Giles and Waterhouse (2009) and Kuo et al. (2015).

Another topic for future research can be the study of higher dimensional models with larger travel times. For the travel times that we considered in this thesis, the particles where not separated in different eddies (see also Figure 7-3) and this is something that we believe will affect the MLMC method. Also, it is interesting to see how the performance of the timestepping methods will be affected after this change.

Finally, another direction for future research could be studying the code of the Met Office's atmospheric dispersion model NAME and obtain a better understanding of how the models are used in practice. The ultimate goal will then be to implement MLMC and our timestepping methods in the NAME model in order to improve its performance. This application will create even more directions for future research since the NAME model also studies problems where the particles interact with each other or are affected by chemical reactions.

Bibliography

Bernardo, M., Budd, C., Champneys, A. R., and Kowalczyk, P. (2008). *Piecewise-smooth Dynamical Systems: Theory and Applications*, volume 163. Springer Science & Business Media.

Bou-Rabee, N. and Owhadi, H. (2010). Long-run accuracy of variational integrators in the stochastic context. *SIAM Journal on Numerical Analysis*, 48(1):278–297.

Cliffe, K., Giles, M. B., Scheichl, R., and Teckentrup, A. L. (2011). Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, 14(1):3–15.

Cook, S. (2013). Multi Level Monte Carlo Methods for Atmospheric Dispersion Modelling. MPhil, University of Bath.

Dacre, H. F., Grant, A. L., Hogan, R. J., Belcher, S. E., Thomson, D., Devenish, B., Marenco, F., Hort, M., Haywood, J. M., Ansmann, A., Mattis, I., and Clarisse, L. (2011). Evaluating the structure and magnitude of the ash plume during the initial phase of the 2010 Eyjafjallajökull eruption using lidar observations and NAME simulations. *Journal of Geophysical Research: Atmospheres*, 116(D20).

Fung, J. C. H. and Vassilicos, J. C. (1998). Two-particle dispersion in turbulentlike flows. *Physical Review E*, 57(2):1677.

George, W. K. (2009). Lectures in Turbulence for the 21st Century. *Chalmers University of Technology*.

Giles, M. and Szpruch, L. (2012). Multilevel Monte Carlo methods for applications in finance. *arXiv preprint arXiv:1212.1377*.

Giles, M. B. (2008). Multilevel Monte Carlo Path Simulation. *OPERATIONS RESEARCH*, 56(3):607–617.

Giles, M. B., Lester, C., and Whittle, J. (2016). Non-nested adaptive timesteps in multilevel monte carlo computations. In *Monte Carlo and Quasi-Monte Carlo Methods*, pages 303–314. Springer.

Giles, M. B., Nagapetyan, T., and Ritter, K. (2015). Multilevel Monte Carlo Approximation of Distribution Functions and Densities. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):267–295.

Giles, M. B., Nagapetyan, T., and Ritter, K. (2017). Adaptive multilevel monte carlo approximation of distribution functions. *arXiv preprint arXiv:1706.06869*.

Giles, M. B. and Waterhouse, B. J. (2009). Multilevel quasi-Monte Carlo path simulation. *Advanced Financial Modelling, Radon Series on Computational and Applied Mathematics*, pages 165–181.

Glasserman, P. (2003). *Monte Carlo Methods in Financial Engineering*, volume 53. Springer Science & Business Media.

Graham, I. G., Scheichl, R., and Ullmann, E. (2015). Mixed Finite Element Analysis of Lognormal Diffusion and Multilevel Monte Carlo Methods. *Stochastic Partial Differential Equations: Analysis and Computations*, pages 1–35.

Hairer, E., Lubich, C., and Wanner, G. (2006). *Geometric numerical integration: structure-preserving algorithms for ordinary differential equations*, volume 31. Springer Science & Business Media.

Heinrich, S. (2001). Multilevel Monte Carlo Methods. In Margenov, S., Waśniewski, J., and Yalamov, P., editors, *Large-Scale Scientific Computing*, volume 2179 of *Lecture Notes in Computer Science*, pages 58–67. Springer Berlin Heidelberg.

Higham, D. J., Mao, X., and Stuart, A. M. (2002). Strong convergence of euler-type methods for nonlinear stochastic differential equations. *SIAM Journal on Numerical Analysis*, 40(3):1041–1063.

Hoel, H., Schwerin, E., Szepessy, A., and Tempone, R. (2012). *Numerical Analysis of Multiscale Computations: Proceedings of a Winter Workshop at the Banff International Research Station 2009*, chapter Adaptive Multilevel Monte Carlo Simulation, pages 217–234. Springer Berlin Heidelberg.

Hoel, H., Von Schwerin, E., Szepessy, A., and Tempone, R. (2014). Implementation and analysis of an adaptive multilevel monte carlo algorithm. *Monte Carlo Methods and Applications*, 20(1):1–41.

Holton, J. (2004). An introduction to dynamic meteorology. *Elsevier Academic Press*.

Jones, A. (2004). Atmospheric dispersion modelling at the Met Office. *Weather*, 59:311–316.

Jones, A., Thomson, D., Hort, M., and Devenish, B. (2007). The U.K. Met Office's Next-Generation Atmospheric Dispersion Model, NAME III. *Air Pollution Modeling and Its Application XVII*, pages 580–589.

Karatzas, I. and Shreve, S. (2012). *Brownian Motion and Stochastic Calculus*, volume 113. Springer Science & Business Media.

Katsiolides, G., Mller, E. H., Scheichl, R., Shardlow, T., Giles, M. B., and Thomson, D. J. (2018). Multilevel monte carlo and improved timestepping methods in atmospheric dispersion modelling. *Journal of Computational Physics*, 354(Supplement C):320 – 343.

Khasminskii, R. (2011). *Stochastic Stability of Differential Equations*, volume 66. Springer Science & Business Media.

Kloeden, P. and Platen, E. (2011). *Numerical Solution of Stochastic Differential Equations*. Stochastic Modelling and Applied Probability. Springer Berlin Heidelberg.

Kuo, F. Y., Scheichl, R., Schwab, C., Sloan, I. H., and Ullmann, E. (2015). Multilevel quasi-Monte Carlo methods for lognormal diffusion problems. *arXiv preprint arXiv:1507.01090*.

Leadbetter, S. J., Hort, M. C., Jones, A. R., Webster, H. N., and Draxler, R. R. (2015). Sensitivity of the modelled deposition of Caesium-137 from the Fukushima Daiichi nuclear power plant to the wet deposition parameterisation in NAME. *Journal of environmental radioactivity*, 139:200–211.

Leimkuhler, B. and Matthews, C. (2015). *Molecular Dynamics: with deterministic and stochastic numerical methods*, volume 39. Springer.

Leimkuhler, B. and Reich, S. (2004). *Simulating hamiltonian dynamics*, volume 14. Cambridge university press.

Lemieux, C. (2009). *Monte Carlo and Quasi-Monte Carlo Sampling*. Springer Science & Business Media.

Lord, G. J., Powell, C. E., and Shardlow, T. (2014). *An introduction to computational stochastic PDEs*. Number 50. Cambridge University Press.

Metropolis, N. and Ulam, S. (1949). The Monte Carlo Method. *Journal of the American Statistical Association*, 44(247):335–341.

Milstein, G. and Tretyakov, M. V. (2005). Numerical integration of stochastic differential equations with nonglobally lipschitz coefficients. *SIAM journal on numerical analysis*, 43(3):1139–1154.

Mueller, E. H., Katsiolides, G., and Shardlow, T. (2017). MLMCLangevin code.

Müller, E. H., Scheichl, R., and Shardlow, T. (2015). Improving multilevel Monte Carlo for stochastic differential equations with application to the Langevin equation. *Proceedings of the Royal Society A*, 471(2176).

Nastrom, G. and Gage, K. S. (1985). A climatology of atmospheric wavenumber spectra of wind and temperature observed by commercial aircraft. *Journal of the atmospheric sciences*, 42(9):950–960.

Øksendal, B. (2003). *Stochastic Differential Equations: An Introduction with Applications*. Springer Berlin Heidelberg.

Ramli, H. M. and Esler, J. G. (2016). Quantitative evaluation of numerical integration schemes for lagrangian particle dispersion models. *Geoscientific Model Development*, 9(7):2441–2457.

Rodean, H. C. (1996). *Stochastic Lagrangian Models of Turbulent Diffusion*. American Meteorological Society.

Shardlow, T. (2006). Modified equations for stochastic differential equations. *BIT Numerical Mathematics*, 46(1):111–125.

Smith, F. and Clark, M. J. (1989). The transport and deposition of airborne debris from the Chernobyl nuclear power plant accident with special emphasis on the consequences to the United Kingdom. Technical report, Meteorological Office.

Taylor, G. I. (1921). Diffusion by continuous movements. *Proc London Math Soc A*, 20:l96–2ll.

Thomson, D. and Devenish, B. (2005). Particle pair separation in kinematic simulations. *Journal of Fluid Mechanics*, 526:277–302.

Thomson, D. J. (1987). Criteria for the selection of stochastic models of particle trajectories in turbulent flows. *Journal of Fluid Mechanics*, 180:529–556.

Wang, F.-Y. and Zhang, X. (2016). Degenerate SDE with Hölder-Dini Drift and Non-Lipschitz Noise Coefficient. *SIAM Journal on Mathematical Analysis*, 48(3):2189–2226.

Watanabe, S. and Ikeda, N. (1981). *Stochastic Differential Equations and Diffusion Processes*, volume 24. Elsevier.

Webster, H., Thomson, D., Johnson, B., Heard, I., Turnbull, K., Marenco, F., Kristiansen, N., Dorsey, J., Minikin, A., Weinzierl, B., Schumann, U., Sparks, R., Loughlin, S., Hort, M., Leadbetter, S., Devenish, B., Manning, A., Witham, C., Haywood, J., and Golding, B. (2012). Operational prediction of ash concentrations in the distal volcanic cloud from the 2010 Eyjafjallajökull eruption. *Journal of Geophysical Research: Atmospheres*, 117(D20).

Webster, H., Thomson, D., and Morrison, N. (2003). New turbulence profiles for NAME. *Met Office Turbulence and Diffusion note 288*.

Wilson, J., Yee, E., Ek, N., and d'Amours, R. (2009). Lagrangian simulation of wind transport in the urban environment. *Quarterly Journal of the Royal Meteorological Society*, 135(643):1586–1602.

Zygalakis, K. C. (2011). On the existence and the applications of modified equations for stochastic differential equations. *SIAM Journal on Scientific Computing*, 33(1):102–130.

Appendix A

A.1 Verifying equation (2.17)

The simplified model is given by

$$dU(t) = -\frac{U(t)}{\tau}dt + \sqrt{\frac{2\sigma_U^2}{\tau}}dW(t). \tag{A.1}$$

This is an Ornstein-Uhlenbeck process so in order to find the solution we follow the standard technique by considering the random variable $Y(t) = e^{\frac{1}{\tau}t}U(t)$.

Applying Itô's formula to $Y(t)$ gives

$$dY(t) = \sqrt{\frac{2\sigma_U^2}{\tau}}e^{\frac{1}{\tau}t}dW(t). \tag{A.2}$$

Since τ and σ_U are constants we assume for simplicity that the system is at equilibrium by setting the start time equal to minus infinity. Integrating equation (A.2) gives

$$Y(t) = \sqrt{\frac{2\sigma_U^2}{\tau}}\int_{-\infty}^{t} e^{\frac{1}{\tau}s}dW(s), \tag{A.3}$$

which implies

$$U(t) = \sqrt{\frac{2\sigma_U^2}{\tau}}e^{-\frac{1}{\tau}t}\int_{-\infty}^{t} e^{\frac{1}{\tau}s}dW(s). \tag{A.4}$$

Therefore, $U(t)$ is Gaussian with mean zero and variance

$$\begin{aligned}
Var(U(t)) &= \frac{2\sigma_U^2}{\tau} e^{-\frac{2}{\tau}t} Var\left(\int_{-\infty}^{t} e^{\frac{1}{\tau}s} dW(s)\right) \\
&= \frac{2\sigma_U^2}{\tau} e^{-\frac{2}{\tau}t} \mathbb{E}\left(\int_{-\infty}^{t} e^{\frac{1}{\tau}s} dW(s)\right)^2 \\
&= \frac{2\sigma_U^2}{\tau} e^{-\frac{2}{\tau}t} \int_{-\infty}^{t} e^{\frac{2}{\tau}s} ds \qquad \text{(Itô isometry)} \\
&= \sigma_U^2.
\end{aligned}$$

Then,

$$\begin{aligned}
\langle U(t)U(t_0)\rangle &= \frac{2\sigma_U^2}{\tau} e^{-\frac{1}{\tau}t} e^{-\frac{1}{\tau}t_0} \langle \int_{-\infty}^{t} e^{\frac{1}{\tau}s} dW(s) \int_{-\infty}^{t_0} e^{\frac{1}{\tau}s} dW(s)\rangle \\
&= \frac{2\sigma_U^2}{\tau} e^{-\frac{1}{\tau}t} e^{-\frac{1}{\tau}t_0} \left[\langle \int_{-\infty}^{t_0} e^{\frac{1}{\tau}s} dW(s) \int_{-\infty}^{t_0} e^{\frac{1}{\tau}s} dW(s)\rangle \right. \\
&\left. + \langle \int_{t_0}^{t} e^{\frac{1}{\tau}s} dW(s) \int_{-\infty}^{t_0} e^{\frac{1}{\tau}s} dW(s)\rangle \right] \\
&= \frac{2\sigma_U^2}{\tau} e^{-\frac{1}{\tau}t} e^{-\frac{1}{\tau}t_0} \left[\langle \left(\int_{-\infty}^{t_0} e^{\frac{1}{\tau}s} dW(s)\right)^2 \rangle \right. \\
&\left. + \langle \int_{t_0}^{t} e^{\frac{1}{\tau}s} dW(s)\rangle \langle \int_{-\infty}^{t_0} e^{\frac{1}{\tau}s} dW(s)\rangle \right] \text{ by independence} \\
&= \frac{2\sigma_U^2}{\tau} e^{-\frac{1}{\tau}t} e^{-\frac{1}{\tau}t_0} \left[\mathbb{E}\left(\int_{-\infty}^{t_0} e^{\frac{1}{\tau}s} dW(s)\right)^2\right] \\
&= \frac{2\sigma_U^2}{\tau} e^{-\frac{1}{\tau}t} e^{-\frac{1}{\tau}t_0} \left(\int_{-\infty}^{t_0} e^{\frac{2}{\tau}s} ds\right) \\
&= \sigma_U^2 e^{-(t-t_0)/\tau}.
\end{aligned}$$

By definition of R

$$\begin{aligned}
R(t-t_0) &= \frac{\langle U(t)U(t_0)\rangle}{\langle U^2(t_0)\rangle} \\
&= \frac{\sigma_U^2 e^{-(t-t_0)/\tau}}{\sigma_U^2} \\
&= e^{-(t-t_0)/\tau},
\end{aligned}$$

which is equal to equation (2.17).

A.2 One-dimensional homogeneous model - Exact solution and particle spread

We now compute the exact solution and an expression of the particle spread for the one-dimensional homogeneous model given by

$$dU(t) = -\lambda U(t)dt + \sigma dW(t), \tag{A.5}$$

$$dX(t) = U(t)dt, \tag{A.6}$$

with $\lambda = \frac{1}{\tau}$ and $\sigma = \sqrt{\frac{2\sigma_U^2}{\tau}}$ positive constants.

A.2.1 Exact solution

For the exact solution we let

$$\mathbf{Y}(t) = \begin{pmatrix} X(t) \\ U(t) \end{pmatrix}, \quad \Lambda = \begin{pmatrix} 0 & 1 \\ 0 & -\lambda \end{pmatrix}, \quad \mathbf{\Sigma} = \begin{pmatrix} 0 \\ \sigma \end{pmatrix} \tag{A.7}$$

so that equations (A.5) and (A.6) can be written as the following Ornstein-Uhlenbeck process

$$d\mathbf{Y}(t) = \Lambda \mathbf{Y}(t)dt + \mathbf{\Sigma} d\mathbf{W}(t). \tag{A.8}$$

Applying Itô's formula to $\mathbf{Z}(t) = e^{-\Lambda t}\mathbf{Y}(t)$ gives

$$d\mathbf{Z}(t) = e^{-\Lambda t}\mathbf{\Sigma} d\mathbf{W}(t). \tag{A.9}$$

Integrating gives

$$\mathbf{Z}(t) = \mathbf{Z}(0) + \int_0^t e^{-\Lambda s}\mathbf{\Sigma} d\mathbf{W}(s), \tag{A.10}$$

which implies that

$$\mathbf{Y}(t) = e^{\Lambda t}\mathbf{Y}(0) + \int_0^t e^{-\Lambda(s-t)}\mathbf{\Sigma} d\mathbf{W}(s). \tag{A.11}$$

The function $e^{\Lambda t}$ is given by

$$e^{\Lambda t} = \sum_{n=0}^{\infty} \frac{(\Lambda t)^n}{n!}, \tag{A.12}$$

and by considering matrix multiplication we can write

$$e^{\Lambda t} = \begin{pmatrix} 1 & \frac{1-e^{-\lambda t}}{\lambda} \\ 0 & e^{-\lambda t} \end{pmatrix}. \tag{A.13}$$

Substituting in equation (A.11) we obtain

$$\mathbf{Y}(t) = \underbrace{\begin{pmatrix} 1 & \frac{1-e^{-\lambda t}}{\lambda} \\ 0 & e^{-\lambda t} \end{pmatrix} \mathbf{Y}(0)}_{A} + \underbrace{\int_0^t \begin{pmatrix} 1 & \frac{1-e^{\lambda(s-t)}}{\lambda} \\ 0 & e^{\lambda(s-t)} \end{pmatrix} \begin{pmatrix} 0 \\ \sigma \end{pmatrix} d\mathbf{W}(s)}_{B}. \tag{A.14}$$

We assume that $\mathbf{Y}(0)$ is independent from the Brownian motion $\mathbf{W}(t)$ and that $X(0)$ is a constant. Therefore, $\mathbf{Y}(t)$ is a random variable with mean $\mathbb{E}[A]$ and variance $Var[A] + Var[B]$.

Computing $\mathbb{E}[\mathbf{Y}(t)]$:

By linearity we have that

$$\mathbb{E}[\mathbf{Y}(t)] = \begin{pmatrix} 1 & \frac{1-e^{-\lambda t}}{\lambda} \\ 0 & e^{-\lambda t} \end{pmatrix} \mathbb{E}[\mathbf{Y}(0)] = \begin{pmatrix} 1 & \frac{1-e^{-\lambda t}}{\lambda} \\ 0 & e^{-\lambda t} \end{pmatrix} \begin{pmatrix} X(0) \\ \mathbb{E}[U(0)] \end{pmatrix}. \tag{A.15}$$

Computing $Var[\mathbf{Y}(t)]$:

For the variable A

$$\begin{aligned} Var[A] &= Var\left[\begin{pmatrix} 1 & \frac{1-e^{-\lambda t}}{\lambda} \\ 0 & e^{-\lambda t} \end{pmatrix} \begin{pmatrix} X(0) \\ U(0) \end{pmatrix}\right] \\ &= \begin{pmatrix} 1 & \frac{1-e^{-\lambda t}}{\lambda} \\ 0 & e^{-\lambda t} \end{pmatrix} Var\left[\begin{pmatrix} X(0) \\ U(0) \end{pmatrix}\right] \begin{pmatrix} 1 & 0 \\ \frac{1-e^{-\lambda t}}{\lambda} & e^{-\lambda t} \end{pmatrix} \\ &= \begin{pmatrix} 1 & \frac{1-e^{-\lambda t}}{\lambda} \\ 0 & e^{-\lambda t} \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 0 & Var[U(0)] \end{pmatrix} \begin{pmatrix} 1 & 0 \\ \frac{1-e^{-\lambda t}}{\lambda} & e^{-\lambda t} \end{pmatrix} \text{ since } X(0) \text{ constant} \\ &= \begin{pmatrix} \frac{(1-e^{-\lambda t})^2}{\lambda^2} & \frac{1-e^{-\lambda t}}{\lambda} e^{-\lambda t} \\ \frac{1-e^{-\lambda t}}{\lambda} e^{-\lambda t} & e^{-2\lambda t} \end{pmatrix} Var[U(0)]. \end{aligned}$$

For the variable B

$$\begin{aligned} Var[B] &= Var\left[\int_0^t \begin{pmatrix} \frac{\sigma}{\lambda}(1-e^{\lambda(s-t)}) \\ \sigma e^{\lambda(s-t)} \end{pmatrix} d\mathbf{W}(s)\right] \\ &= \int_0^t \begin{pmatrix} \frac{\sigma}{\lambda}(1-e^{\lambda(s-t)}) \\ \sigma e^{\lambda(s-t)} \end{pmatrix} \begin{pmatrix} \frac{\sigma}{\lambda}(1-e^{\lambda(s-t)}) & \sigma e^{\lambda(s-t)} \end{pmatrix} ds \\ &= \begin{pmatrix} \frac{\sigma^2}{2\lambda^3}(2\lambda t - 3 + 4e^{-\lambda t} - e^{-2\lambda t}) & \frac{\sigma^2}{2\lambda^2}(1-2e^{-\lambda t}+e^{-2\lambda t}) \\ \frac{\sigma^2}{2\lambda^2}(1-2e^{-\lambda t}+e^{-2\lambda t}) & \frac{\sigma^2}{2\lambda}(1-e^{-2\lambda t}) \end{pmatrix} \end{aligned}$$

By sampling from the distribution of the random variable $U(0)$ and by knowing the mean and covariance matrix of the normally distributed random variable B we have the exact solution of the homogeneous model.

A.2.2 Particle spread

As described in Section 7.1.4 the particle spread is given by the standard deviation of X_t i.e. $\sqrt{Var[X(t)]}$. From Section A.2.1 we have

$$Var[X(t)] = \frac{(1-e^{-\lambda t})^2}{\lambda^2} Var[U(0)] + \frac{\sigma^2}{2\lambda^3}(2\lambda t - 3 + 4e^{-\lambda t} - e^{-2\lambda t}), \qquad \text{(A.16)}$$

and by taking the square root we have an expression for the particle spread with respect to time. Next, we analyse how the short and long time behaviour of the spread is affected by the choice of $U(0)$ which can be deterministic or random.

Deterministic $U(0)$

When $U(0)$ is deterministic we have that $Var[U(0)] = 0$ which gives

$$Var[X(t)] = \frac{\sigma^2}{2\lambda^3}(2\lambda t - 3 + 4e^{-\lambda t} - e^{-2\lambda t}). \qquad \text{(A.17)}$$

For the short time behaviour we Taylor expand equation (A.17) to obtain

$$Var[X(t)] = \frac{\sigma^2}{3} t^3 + higher\ order\ terms, \qquad \text{(A.18)}$$

and for the long time behaviour we neglect the terms $e^{-\lambda t}$ and $e^{-2\lambda t}$ in equation (A.17) to obtain

$$Var[X(t)] \approx \frac{\sigma^2}{2\lambda^3}(2\lambda t - 3). \tag{A.19}$$

Therefore we have

$$\text{spread} \sim \begin{cases} t^{\frac{3}{2}} & \text{for small } t, \\ \sqrt{t} & \text{for large } t. \end{cases}$$

Random $U(0)$

When $U(0)$ is random we assume that $U(0) \sim Normal(0, \sigma_U^2)$ which gives

$$Var[X(t)] = \frac{(1 - e^{-\lambda t})^2}{\lambda^2}\sigma_U^2 + \frac{\sigma^2}{2\lambda^3}(2\lambda t - 3 + 4e^{-\lambda t} - e^{-2\lambda t}). \tag{A.20}$$

As before, for the short time behaviour we Taylor expand equation (A.20) to obtain

$$Var[X(t)] = \sigma_U^2 t^2 + higher\ order\ terms, \tag{A.21}$$

and for the long time behaviour we neglect the terms $e^{-\lambda t}$ and $e^{-2\lambda t}$ to obtain

$$Var[X(t)] \approx \frac{\sigma_U^2}{\lambda^2} + \frac{\sigma^2}{2\lambda^3}(2\lambda t - 3). \tag{A.22}$$

Therefore we have

$$\text{spread} \sim \begin{cases} t & \text{for small } t, \\ \sqrt{t} & \text{for large } t. \end{cases}$$

A.3 A Taylor series expansion approach for the weak convergence of the Geometric Langevin method

In this section we prove the equivalent of Corollary 4.18 for the Geometric Langevin method.

Corollary A.1. *If the Euler-Maruyama method applied to SDEs* (2.7) *and* (2.8) *converges with a weak error of order* 1 *then the weak error of the Geometric Langevin method, applied to the same equations, also converges with at least order* 1.

$Proof.$ As before we denote by $(X_1^{\text{GL}}, U_1^{\text{GL}})$ the Geometric Langevin approximation to the exact solution $(X(h), U(h))$ of the inhomogeneous model after one timestep as described by equations (3.8), (3.9) and (3.10). This approximation can also be written as

$$U_1^{\text{GL}} = e^{-\lambda(X_0)h}U_0 + \sigma(X_0)\alpha_h\xi_0 - \frac{\partial V}{\partial X}(X_0, e^{-\lambda(X_0)h}U_0 + \sigma(X_0)\alpha_h\xi_0)h, \quad \text{(A.23)}$$

$$X_1^{\text{GL}} = X_0 + U_1^{\text{GL}}h, \quad \text{(A.24)}$$

with $\alpha_h = \sqrt{\left(1 - e^{-2\lambda(X_0)h}\right)/2\lambda(X_0)}$ and $\xi_0 \sim Normal(0, 1)$.

We begin by expanding some of the above functions. For $e^{-\lambda(X_0)h}$ we have

$$e^{-\lambda(X_0)h} = 1 - \lambda(X_0)h + \mathcal{O}(h^2). \quad \text{(A.25)}$$

For α_h we have

$$\begin{aligned}
\alpha_h &= \sqrt{h - \lambda(X_0)h^2 + \frac{2}{3}\lambda^2(X_0)h^3 + \mathcal{O}(h^4)} \quad \text{by Taylor expanding the numerator,} \\
&= \sqrt{h}\sqrt{1 - \lambda(X_0)h + \frac{2}{3}\lambda^2(X_0)h^2 + \mathcal{O}(h^3)}, \\
&= \sqrt{h}\left(1 - \lambda(X_0)h + \frac{2}{3}\lambda^2(X_0)h^2 + \mathcal{O}(h^3)\right)^{\frac{1}{2}}, \\
&= \sqrt{h}(1 + \mathcal{O}(h)) \quad \text{by using the binomial theorem.}
\end{aligned}$$

And finally, for the $\frac{\partial V}{\partial X}$ term

$$\frac{\partial V}{\partial X}(X_0, e^{-\lambda(X_0)h}U_0 + \sigma(X_0)\alpha_h\xi_0) = \frac{\partial V}{\partial X}(X_0, (1 - \lambda(X_0)h)U_0 + \sigma(X_0)\sqrt{h}\xi_0 + A_1),$$

where A_1 contains terms which are either constant multiples of $h^{\frac{3}{2}}\xi_0$ or of the order $\mathcal{O}(h^2)$. Then, by Taylor expanding with respect to the second argument and multipying by h we obtain

$$h\frac{\partial V}{\partial X}(X_0, e^{-\lambda(X_0)h}U_0 + \sigma(X_0)\alpha_h\xi_0) = h\frac{\partial V}{\partial X}(X_0, U_0) + A_2, \quad \text{(A.26)}$$

where A_2 has the same form of lead order terms as A_1.

Substituting in equations (A.23), (A.24) they become

$$U_1^{\mathrm{GL}} = (1 - \lambda(X_0)h)U_0 + \sigma(X_0)\sqrt{h}\xi_0 - \frac{\partial V}{\partial X}(X_0, U_0)h + A_3 = U_1^{\mathrm{EM}} + A_3, \tag{A.27}$$

$$X_1^{\mathrm{GL}} = X_0 + U_0 h + A_4 = X_1^{\mathrm{EM}} + A_4, \tag{A.28}$$

where $(X_1^{\mathrm{EM}}, U_1^{\mathrm{EM}})$ is the corresponding Euler-Maruyama approximation and A_3, A_4 are of the same form as A_1, A_2.

We then consider the Taylor expansion of $\phi(X_1^{\mathrm{GL}}, U_1^{\mathrm{GL}})$ where ϕ is a polynomial as in Theorem 4.17.

$$\begin{aligned}
\phi(X_1^{\mathrm{GL}}, U_1^{\mathrm{GL}}) &= \phi(X_1^{\mathrm{EM}} + A_4, U_1^{\mathrm{EM}} + A_3) \\
&= \phi(X_1^{\mathrm{EM}}, U_1^{\mathrm{EM}} + A_3) + \frac{\partial \phi}{\partial X}(X_1^{\mathrm{EM}}, U_1^{\mathrm{EM}} + A_3)A_4 + \mathcal{O}(h^3) \\
&= \phi(X_1^{\mathrm{EM}}, U_1^{\mathrm{EM}}) + \frac{\partial \phi}{\partial U}(X_1^{\mathrm{EM}}, U_1^{\mathrm{EM}})A_3 + \frac{\partial \phi}{\partial X}(X_1^{\mathrm{EM}}, U_1^{\mathrm{EM}} + A_3)A_4 \\
&\quad + \mathcal{O}(h^3).
\end{aligned}$$

All terms except from $\phi(X_1^{\mathrm{EM}}, U_1^{\mathrm{EM}})$ are either constant multiples of ξ_0 or of the order $\mathcal{O}(h^2)$ (since ϕ is a polynomial) so their expectation is either 0 or of the order $\mathcal{O}(h^2)$. Therefore we have,

$$\mathbb{E}[\phi(X_1^{\mathrm{GL}}, U_1^{\mathrm{GL}})] = \mathbb{E}[\phi(X_1^{\mathrm{EM}}, U_1^{\mathrm{EM}})] + \mathcal{O}(h^2). \tag{A.29}$$

As in Definition 4.11 we consider the following difference or bias error

$$\begin{aligned}
\left|\mathbb{E}\left[\phi(X(h), U(h))\right] - \mathbb{E}\left[\phi(X_1^{\mathrm{GL}}, U_1^{\mathrm{GL}})\right]\right| = \; &| \, \mathbb{E}\left[\phi(X(h), U(h))\right] - \mathbb{E}\left[\phi(X_1^{\mathrm{EM}}, U_1^{\mathrm{EM}})\right] \\
&+ \mathcal{O}(h^2) \, |.
\end{aligned}$$

Since we assume that the Euler-Maruyama method converges weakly with order 1 we have

$$\left|\mathbb{E}\left[\phi(X(h), U(h))\right] - \mathbb{E}\left[\phi(X_1^{\mathrm{EM}}, U_1^{\mathrm{EM}})\right]\right| \leq C^{\mathrm{EM}} h^2, \tag{A.30}$$

for some constant $C^{\mathrm{EM}} > 0$ which implies

$$\left|\mathbb{E}\left[\phi(X(h), U(h))\right] - \mathbb{E}\left[\phi(X_1^{\mathrm{GL}}, U_1^{\mathrm{GL}})\right]\right| = \mathcal{O}(h^2), \tag{A.31}$$

and completes the proof by using Theorem 4.17.

A.4 Number of samples on each MLMC level for the one-dimensional inhomogeneous model

In Table A.1 we present some data on the parameters and number of samples per level for two representative MLMC runs, one for the mean particle position and one for the smoothed indicator function. For each timestepping method we choose a tolerance ϵ and present the corresponding number of timesteps M_0 on the coarsest level, the total number of levels L, the timestep size $h_L = \frac{T}{M_0 2^L}$ on the finest level and the number of samples N_l on each level l.

	Mean particle position			Concentration using $P_{[a,b]}^{r=4,\delta=0.1}(X(T))$		
	SE	GL	BAOAB	SE	GL	BAOAB
ϵ	$6.9 \cdot 10^{-5}$	$8.8 \cdot 10^{-5}$	$8.8 \cdot 10^{-5}$	$6.9 \cdot 10^{-5}$	$8.8 \cdot 10^{-5}$	$8.8 \cdot 10^{-5}$
M_0	40	40	40	80	80	40
L	8	4	2	7	3	2
h_L	$9.8 \cdot 10^{-5}$	$1.6 \cdot 10^{-3}$	$6.3 \cdot 10^{-3}$	$9.8 \cdot 10^{-5}$	$1.6 \cdot 10^{-3}$	$6.3 \cdot 10^{-3}$
N_0	6787563	3899945	3432656	68039360	37703814	43656860
N_1	843711	357389	305644	10473620	5864870	11875896
N_2	190585	133744	113927	3220810	2197495	4735493
N_3	55965	48913		1055256	788799	
N_4	18142	17103		361439		
N_5	6064			126656		
N_6	2016			44076		
N_7	748			15776		
N_8	266					

Table A.1: MLMC parameters and number of samples per level for the mean particle position and concentration.